# Re-identification and cross-domain adaptability

Fabian Dubourvieux

Normandie Université

# THÈSE

## Pour obtenir le diplôme de doctorat

**Spécialité Computer sciences**

**Préparée au sein de INSA ROUEN NORMANDIE**

# Re-identification and cross-domain adaptability


## Présentée et soutenue par

## FABIAN DUBOURVIEUX


**Thèse soutenue publiquement le 31/08/2022**
**devant le jury composé de**

| | | |
|---|---|---|
| Mme Alice CAPLIER, | Professeure à l'Université de Grenoble Alpes | Rapporteure |
| M. Michel CRUCIANU, | Professeur au CNAM, Paris | Rapporteur |
| M. Attila BASKURT, | Professeur à INSA de Lyon | Examinateur |
| M. Fabien MOUTARDE, | Professeur à l'école des Mines Paristech | Examinateur |
| M. Stéphane CANU, | Professeur de l'INSA-Rouen, Normandie | Directeur de thèse |
| Mme Samia AINOUZ, | Professeure de l'INSA-Rouen, Normandie | Codirectrice de thèse |
| M Romaric AUDIGIER, | Chercheur scientifique au CEA, Saclay | Membre invité |
| Mme Angélique LOESCH, | Chercheuse scientifique au CEA, Saclay | Membre invité |

**Thèse dirigée par STÉPHANE CANU et Samia AINOUZ**

INSA ROUEN NORMANDIE

MIIS

litis

# Contents

# Chapter 1

# Introduction

## Contents



## 1.1 Re-identification: context and challenges

### 1.1.1 Definition of re-identification and real-world applications

*Re-identification* (re-ID) consists in matching observations of the same individual or object. It can be made from various signals containing information that describes the individual or the object. This thesis focuses more particularly on re-ID using images, for computer vision. As an image understanding task, re-ID differs

from the well-known image *classification*. While classification aims at discriminating between a set of semantic classes of interest, re-ID discriminates between different instances from the same class of interest. Classification discriminates between horses and cows, Fine-grained classification discriminates between different races of cows, while re-ID discriminate between cow "1" and cow "2".
re-ID is therefore considered as an image interpretation task, and has many applications in the real world. The growing number of images to be interpreted translates into a need for automation of image understanding applications such as re-ID.

**Visual authentication and identification for security systems**

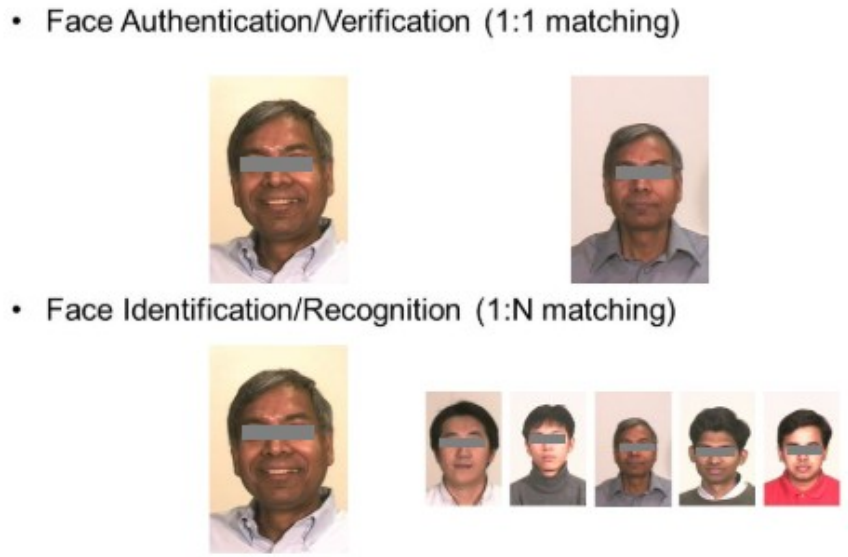


Figure 1.1: Illustration from [60] of two different face-based re-ID applications for authentication and identification: Face Verification and Face Recognition. Face Verification aims at matching the user's face with a face reference in the database. Face Recognition aims at matching the user's face with one of the face references, to assign a specific identity to the user .

Many security systems are based on visual authentication and identification of users. Thus, authentication is successful if the individual described by the photo provided to the system, has allowed its re-identification, i.e. has been matched with a prior reference description given by the security system. For authentication, the re-ID leads to a binary decision, which is also called *verification*. The identification part assigns an identity to the user, which corresponds to that of the reference with which it has been associated by the re-ID. re-ID can also be found in

other practical applications and in other forms. When pictures of the user's face is used for authentication or identification, it is called face verification or recognition [100, 129, 128, 99, 169, 170] (cf Fig. 1.1). These authentication and identification systems based on facial recognition generally aims at extracting biometric-based features from the user's input, that is to say measurable physiological characteristics related to the body, that can uniquely identify an individual while being robust for a a sufficient long period of time for the security system. These biometric-based feature are likely to be related to facial skull measurements, such as the pupillary distance (eye-to-eye distance). Conversely, using soft-biometrics, such as haircuts, could change from one day to the next for the same individual, and degrade the desired long-term robustness of the authentication security system.

**Multi-camera tracking of multiple targets applications.**

Multi-camera tracking relates to many real-world applications. For example, in the context of home assistance, the monitoring of elderly or sick people in a smart-home requires their continuous follow-up inside this place. Using spatio-temporal constraints within the flux of raw images captured by a camera, one or multiple targets of interest can be detected, localized and followed through the time. However, these spatio-temporal constraints can be broken by various practical contexts that break the target tracking:

- Occlusions that can hide a part or the whole target
- Cameras with distinct vision fields which violates spatial continuity assumptions of tracking

When these events occur, it is difficult to ensure that "track 1" that followed "target 1", is still following the "target 1" after the break event. That's where re-ID can be used to match instances before and after these breaking events, in order to correctly reassign the tracks to their corresponding target. More specifically, the re-ID part needs to extract identity-discriminative feature from the bounding boxes given by the tracks before and after the occurring events, and to use these features to match the tracks, as illustrated on Fig. 1.2. These features are expected to be able to discriminate the different targets belonging to the class of interest (people, objects, animals, ...). Contrary to face recognition based authentication, these kind of features should be robust during a shorter time scale conditioned by the breaking event.

Therefore, they are more likely to correspond to soft-biometrics, i.e. features that corresponds to a visual description of the appearance, close to the one a human could give to differentiate different instances (for example semantic attributes such as hair color, type of clothes,... to describe a person).

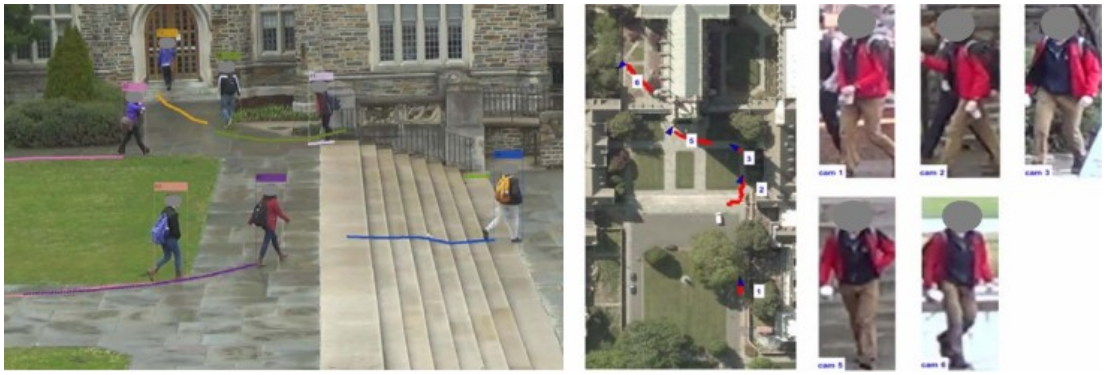

Figure 1.2: The figure from https://reid-mct.github.io/ illustrates a Multi-camera tracking scenario in a network of cameras with disjoint filed of views. Multiple pedestrian can be tracked by the system within a camera view. Between 2 camera views, re-ID is therefore used to match the tracks source.

**Database search applications.**

Another line of important applications involving re-ID are Database search applications. For video-surveillance, a lot of raw images from the video sequences of camera networks are recorded and saved. A user may want to find all the images representing an instance of interest, in this huge flow of recorded data by different camera, at different times. For this, the user provide to the system a *query*, that can correspond to an image of it. The re-ID system here has to match the query to a set of candidate images, called the *gallery*, composed of detection of person or objects within the camera network. Then the re-ID system should provide a ranking of the most relevant images, i.e. the ones that are more likely to correspond to the instance of interest. In this application, re-ID is formulated as a Content-based Image Retrieval (CBIR) [21, 36]. In this CBIR task, re-ID directly characterizes the notion of content, as corresponding to the identity of instances. The re-ID system performs retrieval by extracting features of the query and gallery images, used to compute a similarity score that induces the ranking. As for other re-ID applications, the extracted features and the similarity measurement are expected to discriminate between images representing different instances, and be invariant for images rep-

resenting the same instance. A person search application is illustrated on Fig. 1.3.


Figure 1.3: Example of Person-search application from [173]. The query can be an image of the person of interest, or a text description of the person's appearance. re-ID is used to match the query with the gallery images, by extracting features and computing similarity scores. The relevant gallery images are then returned by the system.

### 1.1.2 Challenges for automatic re-identification

Regardless of the applications, the difficulty for automatic re-ID is to be able to extract from the images the relevant identity information related to the object or individual of interest. This information must be sufficient to discriminate between different identities, while being sufficiently robust to different variability factors to retrieve the different observations of the same instance. As it has been previously mentioned, this relevant information for re-ID depends on the desired application. In the case of a security system by authentication/identification, biometric-like features are desired for their long-term robustness. These biometric feature may be hard to extract for other applications, because the image input does not contain this information or due to privacy constraints. For instance, in the case of public video surveillance, it seems intuitively more difficult to extract enough biometric-related information from the cameras, that could be sufficiently discriminating the individuals. It would therefore be preferable to extract soft-biometric related information from these images, information related to the visual appearance, allowing re-identification of instances over a shorter period than the use of biometric information. Mostly motivated by the videosurveillance and human monitoring applications, computer vision researchers mostly focus on person re-ID, looking for appearance-based features that can discriminate different individuals.

### 1.1.3 Learning re-ID features by supervised learning: limits

**Learning to extract features for re-ID.** To extract the desired identity-discriminative features, re-ID has recently been approached using supervised machine learning, and more specifically representation learning. More specifically, it consists in learning (or inferring) the parameters of a feature extractor (generally a Convolutional Neural Network (CNN) [112]) using a collected dataset composed of raw bounding-boxes of detected instances, manually annotated with an identifier label (ID label). The goal for this feature encoder is being able to extract identity-discriminative features directly from the raw bounding-boxes. Then a similarity score can be computed using these extracted features to perform re-ID. During the training, the feature encoder parameters are inferred only with a subset of samples.

**Supervised learning and the lack of labeled data.** These samples are considered are considered as i.i.d realizations of a data distribution (*domain*). This domain can therefore generate data that are not included in the training data set. It is therefore important that the feature encoder can generalize to unseen data from this domain, that will be encountered when the re-ID feature encoder is deployed. For the re-ID problem, the generalization problem is twofold. The re-ID feature encoder, when used for re-ID in the real-world (deployed), is likely to encounter new images, but also new instances. It means that contrary to the classical classification problems, where the classes labels seen with the training data cover all the classes of interest that will be seen at test time, the instances identifier labels seen at training time for re-ID can be distinct from the instances at testing time. re-ID is therefore called an open-set problem, and this aspect of re-ID mostly motivates the computer vision community to design specific learning techniques for re-ID problem, beyond the direct application of classification methods not adapted to this open-set problem. This requires not overfitting the training set. Preventing overfitting can require having enough samples to train the feature extractor, particularly deep learning architectures with a high number of parameters, such as Convolutional Neural Networks (CNN) [65] that perform well for image understanding tasks. In the case of re-ID, the manual annotation of the ID labels is laborious, time-consuming and therefore costly. Public labeled datasets are therefore generally 100 to 1000 times smaller than large-scale labeled datasets such as ImageNet [25], that lets achieve satisfying classification accuracy using CNN models. To alleviate for this lack of re-ID labeled data, re-ID models rely on *transfer learning*, by fine-tuning a re-ID

feature encoder initialized with one trained for classification on ImageNet.

**The problem of cross-domain performance drop.** Moreover, Supervised Learning works under the assumption that the learned model will be used on images coming from the same domain as the data domain. However, in the real world, the test domain can change from the data training domain. Using the re-ID model on images captured by a different camera than the ones used during training, or on cameras set in a new place, shifts the test domain from the training data domain. As the model specialized on the training domain, a domain shift of the test domain generally causes a drop of performance compared to the training domain. For example, a person re-ID feature extractor trained with data collected outdoor during the summer, in front of a market, is completely ineffective if deployed outdoor during the winter. This is what we observe by training such a re-ID model with a state-of-the-art method on the academic dataset Market-1501 (training data domain), which we then evaluate on the academic dataset Duke-MTMC-reID (test domain) [27]. This reflects a major reliability problem of person of re-ID systems trained with supervised learning. This specialization to the training data domain sets other important practical constraints. Indeed, in practice, re-ID applied to people is subject to ethical rules that ensure the preservation of the people privacy. In France, The Commission Nationale de l'Informatique et des Libertés (CNIL), is in charge of defining these rules and the limits of videosurveillance to preserve privacy. The law for instance imposes a maximum duration for the conservation of images from video surveillance cameras by companies[1]. In particular, the data can require the anonymization of individuals in the collected data. A solution could be the use of computer-generated synthetic data for training the feature extractor. In addition to respecting privacy, it would be a solution without annotation cost. However, the problem of domain shift prevents a system trained on such data from being efficient on images from real cameras. Supervised re-ID is therefore faced with a double problem, combining the domain shift performance drop, the annotation cost and private data availability.

[1] `https://www.cnil.fr/fr/videosurveillance-videoprotection`

## 1.2 A brief review of existing solutions in computer vision literature.

Historically, re-ID has mostly been studied for person re-ID, motivated by pedestrian video-surveillance and the release of public academic labeled datasets for this problem. The re-ID features are extracted and learned from pre-detected pedestrian bounding-boxes images of raw video sequences captured by a set of cameras.

**Handcrafted features.** Early computer vision approaches use handcrafted features for person re-ID [46, 34, 145, 75, 94, 49]. These handcrafted features are designed to discriminate between individuals and invariant to visual factor of variations: backgrounds, colorimetry, poses... These features can be computed directly from the images without the need of data. As for the face recognition problem, these handcrafted features correspond to low-level information, such as edges, shapes or colours. For instance, the Histograms of Gradients features compute the distribution of gradient directions with an histogram, using the image pixel values. These features can be relevant for re-ID since they're designed to be robust to translations, scaling and photometric variations. The first learning-based methods focus on metric learning, by inferring with a set of labeled images, a distance matrix projection to compute image similarity scores [64, 174, 141, 55, 76, 151].This distance matrix projection generally takes as input handcrafted features designed in early re-ID approaches. Metric learning methods therefore focus on improving the similarity measurement for re-ID given a set of features. They do not directly work on the feature extraction part. Some feature extraction algorithm used for re-ID, like Scale-Invariant Feature Transform, can use a set of training images as references to compute the features.

**Unsupervised Learning.** There are also learning-based approaches that do not require any label for images: *Unsupervised Learning* methods (Unsupervised re-ID) [63, 88, 164, 147, 33, 136, 88, 146, 152, 69]. The general principle behind these methods is to estimate and predict the identity labels by the model itself.

**Domain Generalization.** Another area of research for re-ID is Domain Generalization. Domain Generalization faces the problem of domain-shift. To do so, it

seeks to learn a model from a labeled dataset (often with few data) that can better generalize to a domain shift. Contrary to the classical framework of supervised learning, Domain Generalization does not seek to maximize the training data domain re-ID performance, from which the data are taken and that is generally limited due to their small number. In re-ID, only one of these approaches exists, based on the use of instance normalization in deep learning architectures of the ResNet family (ResNet IBN [132]).

**Unsupervised Domain Adaptation.** Another way to tackle the domain shift challenge is Unsupervised Domain Adaptation (UDA). The principle is to exploit labeled data coming from a *source domain*, and to try to exploit the knowledge drawn from this source-domain dataset to learn an effective re-ID model on a different domain of interest called *target domain*. For this, unlabeled data is also available from the target domain. UDA methods for re-ID focus on image-to-image translation techniques [28, 139, 178, 6] and learning features constrained to be domain invariant [118, 180] to reduce the domain gap between the source and target.

## 1.3 Cross-domain adaptability for re-ID

Handcrafted features are completely data-free. They therefore do not overfit the training data domain, since no data is used to compute them. Conceptually these features are designed to be universal in the sense that do not depend on the data distribution. Early metric learning methods do not need as many labeled data as supervised deep learning models.

Unsupervised deep learning models do not need any label for the data, and therefore seem interesting for its annotation cost. Domain Generalization and UDA suppose access to a labeled set, but do not need more annotations to tackle the re-ID domain shift challenge (no more data for Domain Generalization, unlabeled data for the target domain for UDA).

However, for all these approaches, the re-ID performances reported on in-domain and cross-domain benchmarks are very low, compared with Supervised re-ID. Indeed, Unsupervised re-ID, Domain Generalization for re-ID and Unsupervised Domain Adaptation are still early research topics for re-ID. However, for classification, UDA shows promising results to tackle the domain shift challenge. re-ID being an open-set task, it is difficult to directly apply classification UDA methods.

Therefore, this thesis work is motivated by the encouraging results of UDA classification. Then the goal is to study cross-domain adaptability, in the context of UDA, to develop new UDA approach for re-ID, a computer vision task with different specificities.

## 1.4 Contributions

### 1.4.1 Structure of the manuscript

This thesis manuscript is structured as follows. Chapter 1 was an introduction to the re-ID task. It introduced the practical problem of cross-domain performance drop of existing re-ID methods based on Supervised Learning. Among the possible main research directions to tackle the cross-domain re-ID challenge, we chose to focus on Unsupervised Domain Adaption (UDA), motivated by the promising results of early UDA re-ID methods.

In Chapter 2, we review the UDA re-ID related work, and analyze the strengths and weaknesses of the different types of approaches: Domain-Translation and Pseudo-Labeling methods. This leads us to focus on pseudo-labeling methods due to their better cross-domain performance compared to Domain-Translation ones. Therefore, the general guideline of this thesis is to seek how to leverage the useful source domain information in pseudo-labeling methods, to improve their cross-domain performance.

To address this issue, in Chapter 3, an intuition-based approach is proposed to exploit the source data. The idea is to show experimentally that it is possible to improve the cross-domain re-ID performance, by learning a pseudo-labeling model using the labeled source data, in addition to the pseudo-labeled target data. The interpretation of the experiments carried out in this chapter, indicates that to benefit from the source data, it is necessary to limit the impact of the source domain bias on the model during the learning process.

The experimental approach shows limitations in order to determine and understand all the general good practices to systematically benefit from the source knowledge. Chapter 4 therefore proposes a theoretical approach to this problem.

This allows us to deduce the general good practices to implement in a pseudo-labeling method in order to consistently benefit from the source knowledge. Moreover,it gives more insight on the role of this source knowledge in improving cross-domain performance. However, pseudo-labeling approaches, in the context of UDA, still face a major issue that limits the reliability of their cross-domain performance in practice: the sensitivity of cross-domain performance to clustering hyperparameters. This issue is reinforced by the difficulty to select them without target label in the context of UDA re-ID.

Chapter 5 therefore proposes HyPASS, a method for automatic selection of these hyperparameters, using the labeled source data. HyPASS is motivated by theoretical developments. Source-guidance, as well as Conditional Domain Alignment of Feature similarities are shown to be essential to automatic selection of clustering hyperparameters that relies on a labeled source validation set. Various cross-dataset expermiments show the effectiveness of HyPASS to improve the cross-domain performance, while providing more reliability in this performance, compared to an empirical choice of these hyperparameters.

Finally, in Chapter 6, this thesis work is put in perspective with respect to the advances of alternative research directions that can deal with the cross-domain re-ID problem (Unsupervised Learning and Domain Generalization for re-ID), allowing us to propose future research directions.

### 1.4.2 Valorization

**Associated publications on academic data**

- F. Dubourvieux, R. Audigier, A. Loesch, S. Ainouz and S. Canu, "Unsupervised Domain Adaptation for Person Re-Identification through Source-Guided Pseudo-Labeling," 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 4957-4964.

- F. Dubourvieux, A. Loesch, R. Audigier, S. Ainouz and S. Canu, "Improving Unsupervised Domain Adaptive Re-Identification Via Source-Guided Selection of Pseudo-Labeling Hyperparameters," in IEEE Access, vol. 9, pp. 149780-149795, 2021.

- Dubourvieux, F., Audigier, R., Loesch, A., Ainouz, S., & Canu, S. (2021). "A formal approach to good practices in Pseudo-Labeling for Unsupervised Domain Adaptive Re-Identification". arXiv preprint, 2021. Accepted and to be published in CVIU.

**Extension to industrial use cases**

- Dubourvieux, F., Lapouge, G., Loesch, A., Luvision, B., & Audigier, R. (2022). "Cumulative Unsupervised Multi-Domain Adaptation for Cattle Re-identification". Computer Vision and Patter Recognition Workshops (CVPRW) on Computer Vision for Animal Behavior Tracking and Modeling (CV4Animals), 2022. In review at IJCV.

- Patent submission on Automatic Source-Guided Selection of Pseudo-Labeling Hyperparameters. In review.

# Chapter 2

# Related Work

## Contents



This section aims at discussing the work done in the context of Unsupervised Domain Adaptation for cross-domain re-ID (Unsupervised Domain Adaptive re-ID or UDA re-ID for short). In order to understand these approaches, it is necessary to understand the supervised learning for re-ID (Supervised re-ID), introduced in

Sec. 2.1 that underpins the UDA re-ID approaches. It is also necessary to understand how these approaches are evaluated, as described in Sec. 2.2. Then we introduce and discuss existing UDA works in Sec. 2.3 in order to draw a conductive thread for this thesis work in Sec. 2.4.

## 2.1 Supervised Deep Learning for re-ID

re-ID can be broken down into two steps:

- Extracting useful features from the image, i.e. feature that can discriminate between the different instances, while being invariant to different observations of the same individual
- Measuring a feature similarity score to match pairs of images

Early re-ID models mostly focus on the feature extraction part, using handcrafted features. These approaches have been improved by learning better feature similarity measurement (*Metric Learning*), by replacing the cosine similarity or L2 Euclidean distance, generally by the Mahalanobis Distance inferred with a training set [64, 174, 141, 55, 76, 151]. Even with Metric Learning, the low-level information extracted by these features, that can correspond to shapes or colors for example, show limited performance for re-ID. After the success of deep learning for image classification, it has been developed for the re-ID task. In Supervised Deep Learning for re-ID, the feature extractor, parametrized by a Convolutional Neural Network (CNN), is now directly learned from the labeled data (Representation Learning). Representation Learning for Supervised re-ID relies on the design and optimization of different type of loss functions detailed hereafter.

### 2.1.1 General notations for Supervised re-ID

The input space, in the case of re-ID the feature or image space, is denoted by $\mathscr{X} \subseteq \mathbb{R}^{n_x}, n_x \in \mathbb{N}$. The output space is denoted by $\mathscr{Y} \subseteq \mathbb{R}^{n_y}, n_y \in \mathbb{N}$. The re-ID feature extractor $f_\theta : \mathscr{X} \longrightarrow \mathscr{Y}$ is parametrized by $\theta \in \mathbb{R}^p$, where $p \in \mathbb{N}$ is the number of learnable parameters. For learning, a labeled dataset of $n \in \mathbb{N}$ samples $\{x_i, y_i\}_{1 \leq i \leq n}$ where $\forall 1 \leq i \leq n, (x_i, y_i) \in \mathscr{X} \times \mathscr{Y}$, represent concretely the training samples composed of the (detected) images and their one-hot encoded identifier label. The feature can be denoted by $\forall 1 \leq i \leq n, f_i = f_\theta(x_i)$. The pairwise label can be defined such as

$\forall 1 \leq i, j \leq n, r_{ij} = 1$ if $y_i = y_j$ (positive pair) and $r_{ij} = -1$ otherwise (negative pair). The similarity score function measures the re-ID similarity between features, and is defined by $s : \mathbb{R}^{n_f} \times \mathbb{R}^{n_f} \longrightarrow \mathbb{R}$. Similarly, $\forall 1 \leq i, j \leq n, s_{ij} = s(f_i, f_j)$ is set. This similarity score is generally chosen as the cosine similarity ($s_{ij} = \frac{f_i^t f_j}{||f_i||_2 ||f_j||_2}$) or the opposite of the Euclidean distances between the feature vectors ($s_{ij} = -||f_i - f_j||_2$)

### 2.1.2 Pairwise-based metric learning

Pairwise-based metric learning is based on loss functions that optimize directly on the relationship between the data.

### 2.1.3 Verification Loss

Verification Loss [176, 71] aims at predicting if a pair of data have correspond to the same instance (positive) or not (negative). This binary classification task is called a verification task. Given $i, j$ such that $1 \leq i, j \leq n$, the verification task aggregates the individual feature vectors $f_i$ and $f_j$ into a pair feature vector $f_{ij}$ of the same dimension. Usually $f_{ij}$ is computed by the element-wise product (Hadamar product) $f_{ij} = f_i \odot f_j$ [176] or the element-wise squared difference $f_{ij} = (f_j - f_i) \odot (f_j - f_i)$. $f_{ij}$ is then used as the input of a binary classifier (a $(n_f \times 2)$ matrix of learnable parameters) trained to predict if the pair is positive or negative. If $p(r_{ij}|f_{ij})$ represents the predicted sigmoid-activated probability of the pair being recognized as $r_{ij}$ (-1 or 1), the Verification Loss can be computed using the cross-entropy formula:

$$\mathrm{L}_{veri}(i, j) = -\log\big(p(r_{ij}|f_{ij})\big). \tag{2.1}$$

For the verification task, $s_{ij} = p(r_{ij}|f_{ij})$ is generally used as the similarity score.

### 2.1.4 Contrastive Loss

Contrary to Verification Loss, Contrastive Loss directly optimizes pairwise relationships, without the need of classifying pairs of data. Therefore, given $1 \leq i, j \leq n$, the similarity score $s_{ij}$ is is directly optimized in the objective without introducing a classifier, so that the similarity between images of the same individual is increased, and decreased for images representing different individuals. The simplest formula-

tion of the Contrastive Loss introduces a margin $m > 0$ and can be formulated as:

$$\mathrm{L}_{con}(i,j) = (1-\delta_{y_i}^{y_j})\{\max(0, s_{ij} - m)\}^2 + \delta_{y_i}^{y_j} s_{ij}^2, \tag{2.2}$$

where $\delta_{y_i}^{y_j} = 1$ if $y_i = y_j$ and $\delta_{y_i}^{y_j} = 0$ otherwise. While this is the simplest formulation of a Contrastive Loss, re-ID models generally use a Constrastive Loss called the Triplet Loss.

**Triplet Loss.**Triplet Loss is a special Contrastive Loss that considers the re-ID task during the training as a ranking problem. This loss is widely used in Supervised re-ID, since re-ID is generally evaluated as a retrieval ranking task. The idea is to learn $f_\theta$ is to build triplets of data (i,j,k) with $1 \leq i, j, k \leq n$ such that $y_i = y_j$ (positive pair) and $y_i \neq y_k$, and to constrain the similarity of the positive pair $s_{ij}$ to be greater than the negative pair's one $s_{ik}$, by at least a pre-defined margin $m > 0$. The Triplet Loss with a margin is computed by:

$$\mathrm{L}_{tri}(i,j,k) = \max(s_{ij} - s_{ik} - m, 0), \tag{2.3}$$

Mining informative triplets during the training is essential to the performance of a re-ID feature extractor trained with the Triplet Loss. Indeed, in practice, the number of easy triplets (such that $\mathrm{L}_{tri}(i,j,k) \approx 0$) increases quickly as the re-ID model learns, and the the learning signal becomes null, resulting in limited feature discriminability. To alleviate this issue, various informative triplet mining strategies have been designed [53, 117, 137, 122]. In general Triplet Loss based methods build their triplets by selecting, given an anchor sample indexed by $i$, the online hardest positive $j$ ($j$ with the lowest $s_{i,j}$ in a batch) and the online hardest negative $k$ ($k$ with the lowest $s_{i,k}$ in a batch) within each training batch of data.

### 2.1.5 Classification-based metric learning.

The Classification Loss is minimized to learn re-ID features that can well separate the identity classes representing the different individuals in the training set. At testing time, where the re-ID system can see new identity classes, the training model classification layer can be discarded: only $f_\theta$ is used to extract features. If $p(y_i|f_i)$ denotes the predicted probability of $f_i$ being classified as class $y_i$ given by the softmax-activated output of the classification layer, then the Classification Loss

$L_{cls}$ can be computed using the cross-entropy formula

$$L_{cls}(i) = -\log(p(y_i|f_i)), \tag{2.4}$$

The Classification Loss can be viewed as a Metric-Learning objective using class proxies. Indeed, the class-wise classification weights can be viewed in the Euclidean feature space as the class references. More concretely, the classification layer W is represented by a ( can be decomposed into a set of $n_y$ class-wise vector references: $W = [W_1, ..., W_{n_y}]$. Therefore, if $y_i$ designates the one-hot encoded label of class $k_i, 1 \leq k_i \leq n_y$, $p(y_i|f_i) = \frac{\exp(W_{k_i}^T f_i)}{\sum_{k=1}^{n_y} \exp(W_k^T f_i)}$. Therefore, minimizing $L_{cls}$ corresponds to minimizing $W_{k_i}^t f_i$ and maximizing $W_k^t f_i$ for any $k \neq k_i$.In other words, the feature embedding of a sample and its corresponding ground-truth label class reference in the classification layer, are pulled together, by increasing their scalar product, which corresponds to their cosine similarity if the vectors are normalized. In the same way, the feature embeddings and the other class references are pushed away by reducing their scalar product, and therefore their cosine similarity if vectors are normalized. As opposed to pairwise metric learning, which optimizes a learning objective based on the data-to-data relationships, Classification-based metric learning can be seen as proxy-based metric learning, with a learning objective based on data-to-proxy relationships. Therefore, contrary to pairwise methods, proxy methods do not need to mine informative pairs or triplet of data for an effective learning. This may explain why Classification-based metric learning is widely used for re-ID, and often combined with Verification Loss or Triplet Loss.

## 2.2 Evaluating the re-ID performance

### 2.2.1 Evaluation protocol.

In the literature, the re-ID is generally evaluated as a retrieval task. In this context, the re-ID system takes a query image $q \in \mathcal{Q}$ from a set of query images $\mathcal{Q}$, and a ranking $R_q$ of a gallery of images $\mathcal{G}$ is returned by the system. To produce $R_q$, the re-ID system computes feature similarity scores with $s$ between the query and all the gallery images by using the trained re-ID feature extractor $f$. In other words, if the $n$ gallery images are indexed $G = [x_1, ..., x_n]$, the returned ranking R is a ranked list $R_q = [x_{\sigma(1)}, ..., x_{\sigma(n)}]$ given by a permutation σ such that:

$$s\big(f(q), f(x_{\sigma(i)})\big) \geq s(f(q), f(x_{\sigma(j)})), \quad \forall i, j, 1 \leq i \leq j \leq n. \tag{2.5}$$

Basically, the returned ranking correspond to gallery images corresponds to the images of the gallery classified by decreasing order of similarity with the query.

### 2.2.2 Evaluation metrics.

To evaluate re-ID as a retrieval task, the relevance of the returned re-ID ranking is generally measure with Cumulative Matching Characteristics (CMC) [135] and mean Average Precision (mAP) [171] the most used metrics.

CMC-$k$ (Rank-$k$) [135] estimates the probability that a correct match appears in the top-$k$ ranked retrieved results. It can be computed the following formula:

$$\mathrm{CMC}_k = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbb{1}_{y_q \in [y_{\sigma(1)}, \ldots, y_{\sigma(k)}]}(q). \tag{2.6}$$

CMC are accurate when a single view of an instance exists in the gallery for each query, since it only considers the rank of the first good match. However, in a real scenario, the gallery is more likely to contain multiple shots of the same instance in a large camera network, and CMC cannot completely reflect the discriminability of a model across multiple cameras.

The mean Average Precision (mAP) [171], measures the average retrieval performance with multiple views per instance. It can be obtained by computing the Average Precision for each query $q$, given by:

$$\mathrm{AP}(q) = \sum_{k=1}^{|\mathcal{G}|} \mathrm{P}rec_q(k)\mathrm{R}ec_q(k). \tag{2.7}$$

where

$$\begin{cases} \mathrm{P}rec_q(k) = \frac{\sum_{i=1}^{k} \mathbb{1}_{y_q = y_{\sigma(i)}}(q)}{k} \\ \mathrm{R}ec_q(k) = \mathbb{1}_{\{y_q = y_{\sigma(k)}\}}(q). \end{cases} \tag{2.8}$$

Then the mAP is computed by averaging the AP for all queries $q \in \mathcal{Q}$:

$$m\mathrm{AP} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathrm{AP}(q). \tag{2.9}$$

mAP is originally widely used to evaluate retrieval systems. For re-ID evaluation as a retrieval task, it can address the issue of two systems performing equally well in searching the first ground truth (same CMC scores), but having different retrieval abilities for other harder matches to retrieve.

### 2.2.3 Datasets for re-ID.

In the literature, re-ID models are generally evaluated for pedestrian and vehicle re-ID, due to their practical interests and the availability of public datasets. Since re-ID is considered separately from the pedestrian or vehicle detection task, the datasets are composed of views of detected instances, using a detection algorithm. Instance views are then manually annotated with an identity label, and separated into training and test sets.

Although several datasets exist for re-ID, some are more interesting and therefore more used for Supervised re-ID. Indeed, these datasets have certain common characteristics which explain their wide use in re-ID benchmarks. First, they contain enough annotated images (at least 10,000) to train or fine-tune deep-learning models with supervised learning. They also realistically represent the open-set challenge of re-ID since instances in the training and test sets are completely different. Moreover, images used to build these datasets have been collected from networks of cameras with distinct fields of view, which further reflect a real-world scenario where re-ID is of interest.


Figure 2.1: Samples from the three commonly-used person re-ID datasets: Market-1501 (Market) [172], DukeMTMC-re-ID (Duke) [111] and MSMT17 (MSMT) [140]. They are composed of person detections, obtained with an automatic detector from the raw camera images. Each image have en identity label, manually annotated by a human operator.

They are three commonly-used person re-ID datasets: Market-1501 (Market) [172], DukeMTMC-re-ID (Duke) [111] and MSMT17 (MSMT) [140]. Some images from these dataset are displayed on Fig. 2.1.

Market-1501 (*Market*) is composed of 32,668 labeled images from 1501 people captured by 6 outdoor cameras. It is divided into a training set of 12,936 images of 751 pedestrians and a test set with 19,732 images of 750 pedestrians different from the training ones.

DukeMTMC-re-ID (*Duke*) contains 36,411 labeled images of 702 IDs taken by 8 outdoor cameras. It is split into a training set with 6,522 images of 702 pedestrians and 19,889 images of 702 other pedestrians for the test set.

MSMT17 (*MSMT*) is a larger dataset, with 126,441 labeled images of 4,101 pedestrians collected by 15 indoor and outdoor cameras. The training set contains 32,621 images of 1,041 pedestrians and the testing set 93,820 images of 3,060 other pedestrians. It is worth noticing that MSMT17 is a much more challenging dataset than the other two: due to the size of its test set, its number of instances and cameras, MSMT17 is the closest dataset to the conditions of a large-scale re-ID system deployment.

For vehicle re-ID, there are 2 commonly used datasets: Vehicle-ID [82] and Veri-776 [86].

Vehicle-ID is composed of 127,210 labeled images from 13,964 detected vehicles captured by 6 outdoor cameras. It is divided into a training set of 113,346 images of 13,164 vehicles and a test set with 13,864 images of 800 vehicles different from the training ones.

Veri-776 (Veri) is composed of 88,749 labeled images from 13,964 detected vehicles captured by 6 outdoor cameras. It is divided into a training set of 113,346 images of 13,164 vehicles and a test set with 13,864 images of 775 vehicles different from the training ones.

The rising interest in cross-domain re-ID induced the creation of synthetic datasets used as samples from the source domain: *PersonX* [123] and *VehicleX* [86]. PersonX and VehicleX are composed of synthetic images generated on Unity with different types of person and vehicle appearances, camera views and occlusions.

The re-ID datasets characteristics are summarized in Tab. 2.1. As it can be no-

Table 2.1: Dataset composition.

| Dataset | # train IDs | # train images | # test IDs | # gallery images | # query images | avg. #query shots per instance | avg #training shots per instance |
|---|---|---|---|---|---|---|---|
| Market ([172]) | 751 | 12,936 | 750 | 16,364 | 3,368 | 4 | 17 |
| Duke ([111]) | 702 | 16,522 | 702 | 16,364 | 2,228 | 3 | 24 |
| PersonX ([123]) | 410 | 9,840 | 856 | 17,661 | 30,816 | 36 | 24 |
| MSMT ([140]) | 1,041 | 32,621 | 3,060 | 82,161 | 11,659 | 4 | 31 |
| Vehicle-ID ([82]) | 13,164 | 113,346 | 800 | 7,332 | 6,532 | 8 | 9 |
| Veri ([86]) | 575 | 37,746 | 200 | 49,325 | 1,678 | 8 | 66 |
| VehicleX ([86]) | 1,362 | 192,150 | N.A. | N.A. | N.A. | N.A. | 141 |

ticed, they have various statistics, whether in terms of number of images, number of instances or number of shots per instance.

## 2.3 Related Work on Unsupervised Domain Adaptive re-ID

Unsupervised Domain Adaptive re-ID seeks to benefit from the re-ID knowledge from a source domain to a target domain. At training time, a labeled training set from the source domain (source training set) and an unlabeled training set from the target domain (target training set) are available. The goal is to maximize the performance on the target domain, and therefore the UDA re-ID model is evaluated with the target test set at test time. This section introduces the related work on UDA re-ID, in order to discuss it and set the direction of this thesis work. This related work can be divided into two main categories: Domain Translation and Pseudo-Labeling methods.

### 2.3.1 Domain-Translation methods

**Image-level domain translation (IT).**

Image-to-Image translation methods are generative approaches based on learning how to transform images from the source domain to the target one, while preserving the class-related information. In the case of re-ID, this class-related information corresponds to the identity information of the instance, i.e. generally its visual appearance. The goal is to use the source images translated into the target domain style (as shown on Fig. 2.2) to learn in a supervised way a re-ID feature encoder for the target domain. The supervision is performed by reusing the original


Figure 2.2: Illustration from [27] of a Pseudo-Labeling framework by clustering: UDAP. On this figure, the Pseudo-Labeling cycle is divided into 3 steps: (i) A feature encoder, previously trained on the labeled source data, extracts features from the target data. (ii) The extracted features are clustered to predict pseudo-labels. (iii) The feature encoder is fine-tuned, by learning re-ID with the pseudo-labeled target data.

source-domain labels, preserved thanks to the identity information conservation constraint.

With Person Transfer Generative Adversarial Network (PTGAN) [139], a CycleGAN [182] is trained to translate the source images to target-style ones, preserving the identity information by penalizing the reconstruction of the segmented person after translation. Another method [6] leverages a computer-generated synthetic dataset of pedestrians, rendered under various lighting conditions as the source dataset, and then use image-to-image translation for domain adaptation. Similarity preserving generative adversarial network (SPGAN) [28] choose to jointly train the CycleGAN and re-ID feature encoder: the feature similarity between the

translated and original images is maximized to preserve the identity information. The Hetero-Homogeneous Learning (HHL) method [178] models the cross-camera intra-domain discrepancy with a StarGAN that enables image-to-image translation between every camera pairs. HHL therefore further reduces the cross-domain discrepancy by enhancing camera invariance on the target domain of the learned re-ID features. An adaptive transfer network [83] factorizes the domain discrepancy into hypothesized prior factors (illumination, resolution, camera view): using a GAN for each of these factors, the feature encoder is trained to be more robust to them. Suppression of Inter-Domain Background (SBGAN) [57] assumes that the cross-domain background changes explain the cross-domain performance drop: SBGAN therefore generates new images with removed background pixels, trying to preserve the useful ID-related information. Instance-Guided Context Rendering (CR-GAN) [17] generates diverse target-style images with various contexts, by designing a dual conditional GAN which uses the unlabeled target images as the instance-guided contexts. Pose disentanglement is also explored with Pose Disentanglement and Adaptation (PDA) [72] to generate more various target-style images.

Image-to-Image translation methods are constrained to keep the appearance information of the source images, so that they can be assigned the same label after style transfer to the target. The style transfer is therefore only done for the low-level characteristics of the images: the luminosity of the cameras, the colorimetry, the backgrounds, etc. However,the Domain Discrepancy generally also affects appearance: people/objects are a priori different between domains for the re-ID, and there may therefore be a more or less important distribution-shift in the appearance information. For example, images from the Market dataset have been collected during summer, whereas Duke's ones during winter. It is therefore expected that pedestrians do not wear the same type of clothes in Market and Duke, meaning that the domain shift also affects high-level appearance information. Therefore, Image-to-Image translation methods alone, because of their appearance conservation constraint, are therefore limited to tackle the re-ID shift.

**Feature-level domain translation (FT).**

Feature-level domain translation reduces the domain discrepancy at the feature level. For this, these methods directly constrain the learned feature space during the training. Transferable Joint Attribute-Identity Deep Learning (TJ-AIDL) [133] and

Multi-task Mid-level Feature Alignment (MMFA) [78] assume that a feature space where semantic attributes can be discriminated is relevant to be domain-invariant. These methods therefore train a re-ID feature encoder constrained to classify a set of labeled semantic attributes for the source dataset. These approaches seek to align the source and target domain feature distributions by penalizing an unsupervised domain discrepancy loss term [13] or learn domain invariant space by domain feature disentanglement [74] [73] sometimes with auxiliary information for supervision (semantic attribute labels, pose labels...). The camera view information is also leveraged as the supervision signal to reduce the domain gap by learning camera-aware domain-shared features with adversarial learning [108].
Similarly to Image-level domain translation methods, feature-based Domain Translation approaches are struggling to deal with the domain-shift related to the appearance information.

As the supervision of these methods is fully based on the source-data, it is questionable whether a Domain-Invariant Feature space where features that are identity-discriminative on the source domain, are also identity-discriminative on the target domain. In other words, it is unknown if the source domain contains enough useful information to only rely on it, in order to learn target re-ID features. These feature-based Domain Translation approaches, by design, can at best extract this useful information: with a source-only supervision, target-specific identity-discriminative information cannot be extracted. For example, in Market, with a dataset captured during summer, no pedestrian wears a coat in the dataset, contrary to Duke captured during winter where it is common. Intuitively, with only supervision on Market, it seems hard to learn discriminative appearance based feature related to pedestrian wearing a coat for Duke, whereas no coat is seen in Market. That's why Feature-based Domain Translation methods with semantic attributes, such as TJ-AIDL, manually and intuitively select a set of semantic attributes shared between the source and target domains.

### 2.3.2 Pseudo-Labeling methods (PL).

Pseudo-Labeling methods consists in mining identity-related supervision for the unlabeled target dataset. The goal is to make the feature representation on the target domain more identity-discriminative to improve the cross-domain re-ID performance. Generally, for this, a re-ID feature encoder trained with the source data

is leveraged. Therefore, different approaches are designed to get ID supervision in Pseudo-Labeling methods.

**Pseudo-Labeling by clustering.**

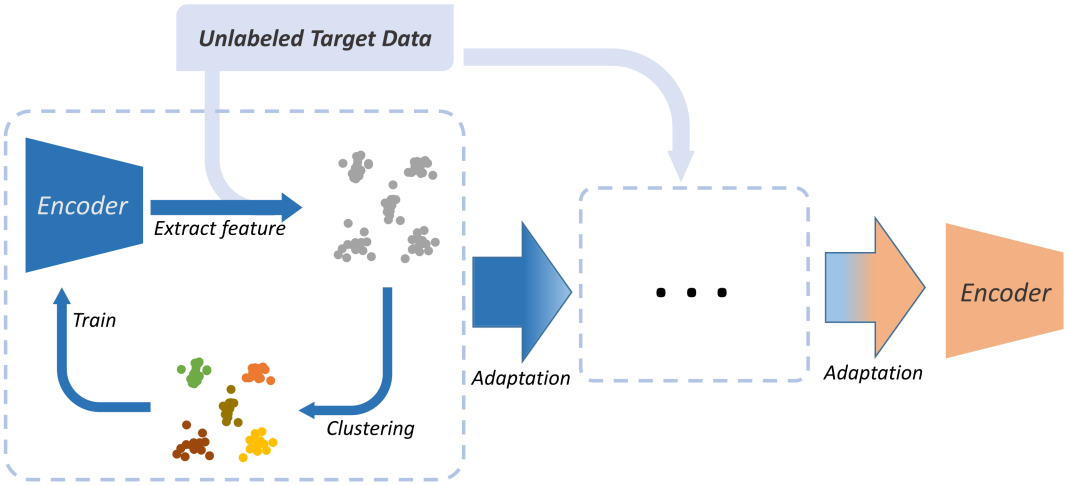


Figure 2.3: Illustration of a Pseudo-Labeling framework by clustering: UDAP. On this figure, the Pseudo-Labeling cycle is divided into 3 steps: (i) A feature encoder, previously trained on the labeled source data, extracts features from the target data. (ii) The extracted features are clustered to predict pseudo-labels. (iii) The feature encoder is fine-tuned, by learning re-ID with the pseudo-labeled target data.

The main line of Pseudo-Labeling methods predict identity pseudo-labels by clustering the target re-ID features. Given that the cross-domain re-ID problem is open-set, the identity classes in the target set are a priori distinct from those in the source training dataset. Therefore, using a classifier trained to separate the source training set ID classes is not relevant to predict these unknown new identity classes. Therefore, works on Pseudo-Labeling re-ID assume that these target ID classes can be assigned by clustering the target features, obtained by a re-ID feature encoder initially trained with the source dataset. The ID pseudo-labels obtained for the target data are then used with the classical supervised learning techniques for re-ID.

Self-training with progressive augmentation framework (PAST) [158] and UDAP [120] (illustrated on Fig. 2.3) are the first methods to design a self-learning paradigm based on Pseudo-Labeling by clustering. It corresponds to an iterative and cyclic training scheme where the pseudo-labels are updated regularly using the re-ID feature encoder trained with the previous pseudo-labels. The main and most recent

line of works on Pseudo-Labeling are built on this general iterative self-training scheme [120, 159, 62, 124, 156, 183, 37, 14, 41, 157, 163, 183, 103, 153, 181, 90]. Generally, the density-based spatial clustering of applications with noise (DBSCAN) algorithm is preferred to predict the pseudo-labels. DBSCAN builds the clusters based on regions of high density in the feature space. The density in DBSCAN is defined by setting two hyper-parameters: the minimum distance between two samples from the same neighborhood, as well as a minimum number of samples in a cluster. Therefore, the samples outside of these regions of high density are considered as noise by the algorithm. Intuitively, DBSCAN allows to discard a number of samples with low clustering-confidence, which is likely to reduce the number of errors in the pseudo-label samples used during the training.

**Pseudo-Labeling by leveraging the temporal information.**

A line of methods focus on using the temporal information to get pseudo-labels for the target data. They generally use a time-frame information with the camera label, assuming that images captured by the same camera, in a sufficiently short amount of time (defined beforehand), correspond to the same individual or object. In the same way, images from different cameras and time segment are considered as representing different individual or object. These temporal rules are defined and used in Unsupervised Camera-aware Domain Adaptation (UCDA) [108] to generate triplet with target images (reference, positive, negative) to perform supervised learning by metric learning. In TFusion [93], these rules on temporal continuity are implemented in a Bayesian framework, to estimate the probability that a pair of images represent the same individual or not.

**Pseudo-Labeling by pairwise supervision.**

The pairwise supervision focuses on mining pairs of the same or different ID for the unlabeled target data. These methods generally define and compute a pairwise similarity score in the feature space that aims at reflecting the ID closeness of the pair of data. Therefore, a high similarity score indicates a high likelihood that the pair corresponds to the same ID, and a low score that they correspond to different ID. To make the feature space more identity-discriminative, similarity scores corresponding to positive pairs should be high, and those corresponding to negative pairs should be low. In the Exemplar Camera Neighborhood (ECN) [180] and

ECN+ [90] methods, the pairwise supervision relies on a definition of neighborhood in the learned feature space. To improve their re-ID robustness on the target domain. Given a target sample, its k-nearest neighbors in the feature space (based on the cosine similarity scores computed between this sample and the other in the target dataset), i.e. the k samples associated to the k highest similarity scores with the reference, are considered as positive pairs with respect to the reference. With these binary positive pseudo-labels defined by the neighborhood, the feature encoder is then trained to increase the similarity scores between neighbors. Multilabel reference learning (MAR) [165] computes a similarity score based on the feature similarity between a target sample and a set of references representing the source sample ID classes. This set of scores with respect to the source references are then used as multi-class soft labels, to predict positive or negative pseudo-labels for pairs of target data.

**Improving Pseudo-Labeling methods.**

Pseudo-Labeling works explore various research directions to improve the target re-ID performance. The main research direction for Pseudo-Labeling approaches focuses on limiting the impact of pseudo-label errors on the final UDA re-ID performance. To this end, various Pseudo-Labeling methods have been designed.

Mutual Mean Teaching (MMT) [41] and Multiple Expert Brainstorming (MEB-Net) propose mutual learning, a learning strategy where a pair of teacher-student networks collaborate with each other to reduce the impact of learning with noisy labels. This robustness is strengthened by Mean Teaching, i.e. the use of temporal moving average of the teacher parameters throughout the learning process. Leveraging multiple cluster views can also improve the robustness to noisy labels [35]. Some works design new learning criteria based on the global distance distributions ([62, 87]), assumed to be more robust to outliers than optimization directly on the distances. Other methods focus on leveraging local features ([38], intra-inter camera features ([142, 80, 144]), or even class-centroid and instance feature memory banks with contrastive learning ([42]). Another line of work focuses on pseudo-label refinement, by using attention-based models ([61]), by combining pseudo-labels with domain-translation/generative methods ([156, 124, 183, 16]), by using online pseudo-labels ([167, 175]) predicted at the batch-level. Label propagation ([160]) is also used to get more consistent pseudo-labels through the training. Other approaches focus on designing efficient sample selection and outlier detection strate-

gies ([37, 14]). Then, a recent line of work seeks to leverage the source knowledge during pseudo-label training ([42, 59]), contrary to other works that discard the source data after the first pseudo-label generation for the target data.

Table 2.2: Performance comparison of UDA Person re-ID state-of-the-art methods on Duke [111] and Market [172] used for cross-dataset benchmarks. mAP and rank-1 accuracy are reported in %. Different colors are associated to the different UDA approach types: IT (Image Translation), FT (Feature Translation), and PL (Pseudo-Labeling).

| Methods | Approach Type | Duke→Market | | Market→Duke | |
|---|---|---|---|---|---|
| | | mAP | rank-1 | mAP | rank-1 |
| UMDL [104] (CVPR'16) | IT | 12.4 | 34.5 | 7.3 | 18.5 |
| PTGAN [140] (CVPR'18) | IT | - | 38.6 | - | 27.4 |
| SPGAN [27] (CVPR'18) | IT | 22.8 | 51.5 | 22.3 | 41.1 |
| ATNet [84](CVPR'19) | IT | 25.6 | 55.7 | 24.9 | 45.1 |
| TJ-AIDL [133] (CVPR'18) | FT | 26.5 | 58.2 | 23.0 | 44.3 |
| CFSM [13] (AAAI'19) | FT | 28.3 | 61.2 | 27.3 | 49.8 |
| UCDA [109] (ICCV'19) | FT | 30.9 | 60.4 | 31.0 | 47.7 |
| HHL [179] (ECCV'18) | IT | 31.4 | 62.2 | 27.2 | 46.9 |
| ARN [74] (CVPR'18-WS) | FT | 39.4 | 70.3 | 33.4 | 60.2 |
| ECN [181] (CVPR'19) | PL | 43.0 | 75.1 | 40.4 | 63.3 |
| PDA-Net [73] (ICCV'19) | FT | 47.6 | 75.2 | 45.1 | 63.2 |
| UDAP [120] (PR'20 / arXiv'19) | PL | 53.7 | 75.8 | 49.0 | 68.4 |
| PCB-PAST [159] (ICCV'19) | PL | 54.6 | 78.4 | 54.3 | 72.4 |
| SSG [37] (ICCV'19) | PL | 58.3 | 80.0 | 53.4 | 73.0 |
| MPLP+MMCL [130] (CVPR'20) | PL | 60.4 | 84.4 | 51.4 | 72.4 |
| ACT [37] (AAAI'20) | PL | 60.6 | 80.5 | 54.5 | 72.4 |
| AD-Cluster [156] (CVPR'20) | PL | 68.3 | 86.7 | 54.1 | 72.6 |
| MMT [41] (ICLR'20) | PL | 71.2 | 87.7 | 65.1 | 78.0 |
| CAIL [91] (ECCV'20) | PL | 71.5 | 88.1 | 65.2 | 79.5 |
| NRMT [163](ECCV'20) | PL | 71.7 | 87.8 | 62.2 | 77.8 |
| B-SNR+GDS-H [62](ECCV'20) | PL | 72.5 | 89.3 | 59.7 | 76.7 |
| MEB-Net [157](ECCV'20) | PL | 76.0 | 89.9 | 66.1 | 79.6 |
| SpCL [42] (NeurIPS'20) | PL | 76.7 | 90.3 | 68.8 | **82.9** |
| Dual-Refinement [24] (TIP'21) | PL | 78.0 | 90.9 | 67.7 | 82.1 |
| UNRN [166] (AAAI'21) | PL | 78.1 | 91.9 | 69.1 | 82.0 |
| GLT [168] (CVPR'21) | PL | **79.5** | **92.2** | **69.2** | 82.0 |

Table 2.3: Different colors are associated to the different UDA approach types: IT (Image Translation), FT (Feature Translation), and PL (Pseudo-Labeling).

| Methods | Approach Type | Market→MSMT | | Duke→MSMT | |
|---|---|---|---|---|---|
| | | mAP | rank-1 | mAP | rank-1 |
| PTGAN [140] (CVPR'18) | IT | 2.9 | 10.2 | 3.3 | 11.8 |
| ENC [181] (CVPR'19) | PL | 8.5 | 25.3 | 10.2 | 30.2 |
| SSG [37] (ICCV'19) | PL | 13.2 | 31.6 | 13.3 | 32.2 |
| NRMT [163] (ECCV'20) | PL | 19.8 | 43.7 | 20.6 | 45.2 |
| CAIL [91] (ECCV'20) | PL | 20.4 | 43.7 | 24.3 | 51.7 |
| MMT [41] (ICLR'20) | PL | 22.9 | 49.2 | 23.3 | 50.1 |
| Dual-Refinement [24] (TIP'21) | PL | 25.1 | 53.3 | 26.9 | 55.0 |
| UNRN [166] (AAAI'21) | PL | 25.3 | 52.4 | 26.2 | 54.9 |
| GLT [168] (CVPR'21) | PL | 26.5 | 56.6 | 27.7 | 59.5 |
| SpCL [42] (NeurIPS'20) | PL | 26.8 | 53.7 | 26.5 | 53.1 |

## 2.4 Related Work Discussion & Thesis direction

The cross-domain re-ID performance of the previously introduced UDA re-ID methods are reported in Tab. 2.2 and Tab. 2.3. The Domain-Translation approaches obtain the worst performance of the state-of-the-art UDA methods. This observation is in line with our criticisms of this type of approach: used alone and as such, they are not sufficient to manage a re-ID domain-shift that also affects the appearance distribution of the instances.

Concerning Pseudo-Labeling approaches, they have been the main research direction for UDA re-ID due to their better performance on the target domain compared to UDA re-ID based on Domain translation. The Pseudo-Labeling methods obtain significantly the best performance in the state of the art. At the beginning of this thesis, the first Pseudo-Labeling approaches were designed and already seemed promising, with performances above Domain Translation. The general direction of researches on Pseudo-Labeling, has been to reduce the impact of errors in pseudo-labels on the training process or improving the quality of the predicted pseudo-labels. While Domain Translation methods are source-based UDA methods, Pseudo-Labeling methods tend to be target-based, by discarding the source data after initialization of the first pseudo-labels. That's why the core direction of this thesis is to investigate how the source knowledge could further be leveraged in

Pseudo-Labeling methods, to tackle the cross-domain re-ID challenge.

# Chapter 3

# An Empirical approach to Source-Guided Pseudo-Labeling

## Contents



Unsupervised Domain Adaptation (UDA) aims at exploiting source knowledge to improve cross-domain performance without annotating target domain data. Although Pseudo-Labeling methods show promising performance for cross-domain re-ID, a review of existing approaches suggest that they may under-exploit labeled source data, that is only used for initializing the first pseudo-labels. This chapter

therefore aims at experimentally finding conditions under which source data can be further leveraged by a Pseudo-Labeling method, in order to improve its cross-domain performance. This motivation is further detailed in Sec. 3.1. Then the goal of Sec. 3.2 is to design such a source-guided pseudo-labeling method for UDA re-ID. For this purpose, we first propose to experimentally evaluate a naive framework inspired by pseudo-labeling UDA for classification, on cross-domain re-ID benchmarks. Result interpretation leads us to assume that the source bias should be reduced during learning to improve the cross-domain performance when using the source data. Then a new Source-Guided pseudo-labeling method for UDA re-ID is proposed, by implementing Domain-specific batch normalization and a Two-head feature encoder. Experiments and ablative study show that this Source-Guided method can indeed improve the cross-domain re-ID performance on pedestrian and vehicle cross-dataset benchmarks. Finally, in Sec. 3.3, we discuss the limits of an empirical approach to design a source-guided pseudo-labeling UDA re-ID.

A paper related to this chapter has been published in an international conference [30].

## 3.1 Motivation

### 3.1.1 Do existing UDA re-ID Pseudo-Labeling methods sufficiently leverage the labeled source data ?

The most successful UDA re-ID methods for cross-domain re-ID are those based on Pseudo-Labeling. This learning process based on several iterations of Pseudo-Labeling cycles. The source dataset is exploited at the initialization stage, i.e. to train in a supervised way the feature encoder $f_\theta$ used to generate the initial pseudo-labels for the target data. Once these pseudo-labels have been generated, Pseudo-Labeling methods generally fine-tune the model learned from the source data. Unlike Domain Translation approaches, the source data is then not used at all during training. Particularly at the time of the work in this chapter in 2019, none of the early Pseudo-Labeling approaches use the source data after the first pseudo-labels have been obtained.

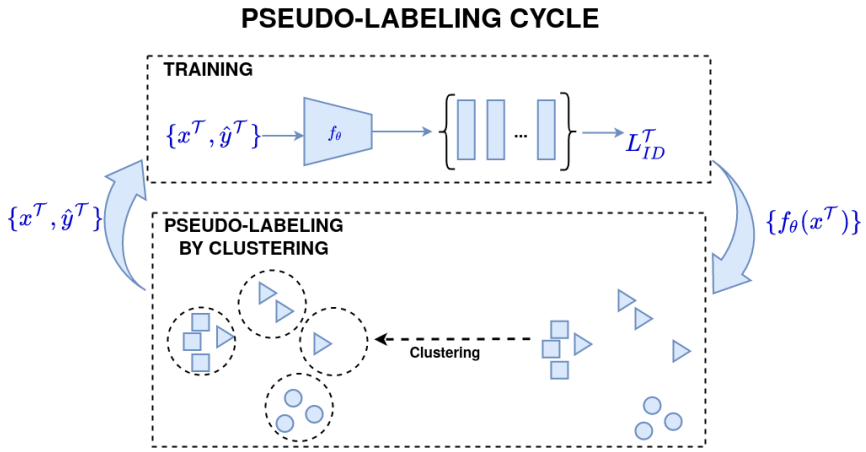


Figure 3.1: Illustration of the classical Pseudo-Labeling cycle. A feature encoder $f_\theta$ trained on the source data, is used to extract features $\{f_\theta(x^{\mathcal{T}})\}$ from the target training set images $\{x^{\mathcal{T}}\}$, which are used to predict pseudo-labels by clustering. $f_\theta$ is then fine-tuned by feature re-ID learning by minimizing a re-ID loss function $L_{ID}^{\mathcal{T}}$ with the pseudo-labeled target data. Pseudo-Labeling methods perform several Pseudo-Labeling cycles to update and improve the pseudo-labels, which subsequently improves the performance of the learned feature encoder.

It can be noted that in the MAR [165] and UDAP [120] approaches, the source data is used in the iterative cycle of Pseudo-Labeling, at the clustering step as illustrated on Fig. 3.1. MAR uses the source data to define the source class representatives that are used to predict the pseudo-labels by computing the similarity to them. UDAP proposes to add the similarity of the target data to the source data in the calculation of the pairwise similarity matrix of the target features, which is used to predict the pseudo-labels in the DBSCAN clustering algorithm. In both cases, the source data is only used for the prediction of the pseudo-labels, and is therefore not directly taken into account to train the model.Overall, the source data is never used during training in Pseudo-Labeling UDA re-ID methods. We therefore propose to leverage the labeled source data during training, in order to improve the cross-domain re-ID performance of Pseudo-Labeling methods. This raises two questions: Do Pseudo-Labeling UDA re-ID methods can benefit from the labeled source data used during training ? If so, how the source data should be used to get this improvement ?

### 3.1.2 How the labeled source data is used in Pseudo-Labeling UDA classification

Many recent Pseudo-Labeling UDA classification methods leverage the labeled source data during the training after generating the target pseudo-labels. The labeled source data and the pseudo-labeled target data are used in a symmetric way, optimizing the same classification losses. Using or discarding the source data for Pseudo-Labeling training are not considered by these methods as a significant part of their design. Therefore, there seems to be no experimental ablative study that would indicate that keeping optimization on the source data is better. However, two pioneer papers might explain why the source is used or discarded for Pseudo-Labeling UDA classification.

#### Optimal Mixing Value in the defense of the source benefit for UDA classification

The first paper is a theoretical work on UDA conducted by Ben David et al. [8]. In their paper, they consider a similar problem called Optimal Mixing Value. Basically, it is an UDA-like problem, where the goal is to maximize the target classification performance. For this, a mix of labeled source and labeled target samples is used for training. While the target samples are supposed being labeled, and not pseudo-labeled, it tackles the key question of the usefulness of the labeled source samples, in addition to the target samples. Ben David et al. [8] theoretically prove that the source samples can help to improve the target classification performance when used in addition to the target samples, under some conditions set by the UDA problem. These conditions are illustrated by a theoretical threshold beyond which the source becomes useful, and where it would therefore be preferable to use only target data. This threshold depends on the number of source and target data in the training set, on the domain gap between the source and target defined in their paper, and of the complexity of the class of models used for learning the classifier (measured by the VC dimension). The difficulty of estimating precisely this VC dimension for classes of models with neural networks makes it even more hard to estimate this threshold for practical purposes. Moreover, the assumptions and problem differ from UDA Pseudo-Labeling, in that Ben David et al. [8] consider ground-truth labels for the target data, instead of pseudo-labels that can be erroneous. Nevertheless, their work may provide a way to show that, even with supervision on the target domain, additional supervision with the labeled source data may be beneficial.

**DIRT-T: in the defense of target-only Pseudo-Labeling UDA classification**

DIRT-T [116] seems to be the pioneer paper introducing Pseudo-Labeling in the UDA framework for classification. The paper highlights two weaknesses of previously used domain alignment approaches. Firstly, the deep neural networks used in these approaches are shown by Shu et al. [116] to be a class of models rich enough to learn feature projections that can easily reduce the domain gap estimated with the training data. Moreover, Shu et al. show that such models, that completely minimize the domain gap on the training data, can give non-discriminative feature representation. In other words, optimizing a feature encoder to reduce the domain gap, is not sufficient to tackle the UDA classification problem.
Secondly, these approaches assume that the domain adaptation problem is conservative, i.e. that a feature encoder, in the class of models considered for learning, can perform well on both domains simultaneously. This assumption cannot be verified, notably due to the absence of labels for the target domain data, and has no reason to be true for a given UDA problem. The performances obtained by these domain alignment approaches could therefore only be limited. To address these limitations, the authors focus on the cluster assumption from semi-supervised learning, which states that the decision boundaries of a classifier should not traverse regions of high density space. The classical UDA approaches, based on Domain Alignement, are supposed according to Shu et al. to learn classifiers that badly respect the cluster assumption on the target domain, due to the previously mentioned limitations. Pseudo-Labeling is introduced for this purpose, inspired by the work of Grandvalet et al. [45] on entropy minimization for semi-supervised learning. Pseudo-labels are thus introduced as a way to promote entropy minimisation during training and consequently to ensure better non-violation of the cluster assumption on the target domain. In DIRT-T, the pseudo-label is formulated as an iterative learning algorithm of the Teacher-Student type, where the Teacher corresponds to the model at a previous iteration, which was used to predict the pseudo-labels used for the optimization of the classification error with the target Student (the current model). Therefore, Shu et al. suggests with DIRT-T that the source is only useful to initialize the Teacher model used to predict the first pseudo-labels. Then, only the pseudo-labeled target samples are useful to fine-tune a Student model, and make its feature representation more discriminative.

**A discussion about the benefit of leveraging the source for Pseudo-Labeling training.**

Works on Optimal Mixing Value from Ben David et al. [8] demonstrated that the source domain can be a valuable knowledge, in addition to target labeled data, to improve the classification performance on the target domain. This suggests that the same could be true for UDA re-ID, and with pseudo-labels for the target domain data. Although DIRT-T [116] originally proposed a target-only Pseudo-Labeling paradigm, in order to further enforce the cluster assumption on the target domain, it does not exclude the possibility that using the source data could further benefit the target domain performance, outside the scope of cluster assumption. Existing UDA re-ID Pseudo-Labeling methods therefore follow the target-only Pseudo-Labeling paradigm in line with DIRT-T.
In line with the theoretical work on Optimal Mixing Value for the closed-set classification task, our goal is to design an approach that mixes both source and target samples during training, but, in our case, that could apply for the open-set UDA re-ID task and that deals with target pseudo-labels.

## 3.2 Designing a Source-Guided Pseudo-Labeling UDA re-ID approach

The goal is to design a general Source-Guided Pseudo-Labeling UDA re-ID approach that can benefit the cross-domain re-ID performance. The challenge is twofold:

- the generability of this Source-Guided approach should allow its integration into any IDA re-ID Pseudo-Labeling method, in order to make it Source-Guided

- the contribution of the source is expected to improve the cross-domain re-ID performance compared to the target-only version of the Pseudo-Labeling method

### 3.2.1 A first empirical approach to Source-Guided Pseudo-Labeling

#### The Naive Source-Guided Pseudo-Labeling method

As a first approach, we naively propose to optimize the loss function proposed by Ben David et al. [8] for Optimal Mixing Value classification, replacing by analogy the classification losses by re-ID losses, and using pseudo-labels for target data instead of ground-truth labels. We call it the Naive Source-Guided Pseudo-Labeling method.

Therefore, inspired by the work of Ben David et al.[8] on classification, a Source-Guided Pseudo-Labeling for UDA re-ID can be designed by using the source samples in a similar way to the target samples. This means that during training, if $L_{ID}^{\mathscr{T}}$ denotes the re-ID loss function used to learn the re-ID feature encoder $f_\theta$ (for example one of the losses function described in Sec. 2.1) evaluated with a set of pseudo-labeled target samples, and $L_{ID}^{\mathscr{S}}$ the same loss function evaluated on a set of labeled source samples, then $f_\theta$ is learned by minimizing the loss function $L_{ID}$ given by:

$$L_{ID}(\theta) = \alpha L_{ID}^{\mathscr{S}}(\theta) + (1-\alpha) L_{ID}^{\mathscr{T}}(\theta), \alpha \in [0,1]. \tag{3.1}$$

$\alpha$ is a hyperparameter that is set before training. In their work, Ben David et al. [8] derive a formula to estimate $\alpha$. However the formula applies only to classification and assumes that the target labels are ground-truth labels. Moreover, for re-ID, deep neural networks are used to learn $\theta$, which makes it difficult to calculate the VC dimension.

In addition, the UDA setting makes it even harder to use model selection with a validation set since no target labels is available for it. Therefore, $\alpha$ is arbitrarily set to 0.5. Hereafter is detailed a general pseudo-code (Alg. 1) of the Naive Source-Guided Pseudo-Labeling. The general Naive Source-Guided Pseudo-Labeling cycle is also illustrated on Fig. 3.2.

#### Experimental settings

**Experimental protocol.** To show that Naive Source-Guidance (Naive SG) can easily be added and contribute to various target-only Pseudo-Labeling UDA methods, it is integrated into two different state-of-the-art target-only methods:

- UDAP [120]: The classical Pseudo-Labeling UDA algorithm for re-ID which is

**Algorithm 1** Naive Source-Guided Pseudo-Labeling

**Input:** Labeled source training data $D^{\mathcal{S}} = (X^{\mathcal{S}}, Y^{\mathcal{S}})$, unlabeled target data $X^{\mathcal{T}}$, clustering algorithm C, an initialization phase re-ID loss function $L_0^{\mathcal{S}}$, a Source-Guided Pseudo-Labeling re-ID loss function $L_{ID}$, a number of Pseudo-Labeling iterations $N_{iter}$, an initial encoder $f_\theta^{(0)}$

1: Train the initial encoder $f_\theta^{(0)}$ on $D^{\mathcal{S}}$ by minimizing $L_0^{\mathcal{S}}(\theta)$
2: Initialize $f_\theta \leftarrow f_\theta^{(0)}$
3: **for** $t = 1$ to $N_{iter}$ **do**
4: Compute target features: $F^{\mathcal{T}} \leftarrow f_\theta(X^{\mathcal{T}})$
5: Pseudo-label the target samples by clustering: $(X^{\mathcal{T}}, Y^{\hat{\mathcal{T}}}) \leftarrow C(F^{\mathcal{T}}, X^{\mathcal{T}})$
6: Train $f_\theta$ during $N_{epoch}$ by minimizing $L_{ID}$ using $D^{\hat{\mathcal{T}}} = (X^{\mathcal{T}}, Y^{\hat{\mathcal{T}}})$ and $D^{\mathcal{S}}$
7: **end for**
8: Return $f_\theta$

not designed to be robust to overfitting pseudo-label errors. It uses DBSCAN [32] to predict the pseudo-labels.

- MMT [41]: the best state-of-the-art Pseudo-Labeling UDA method at the time of this work. It mitigates for bad effects due to pseudo-label errors by combining Mutual Learning [161] and Mean Teaching [125] known in Semi-Supervised Learning. It uses k-means to generate the target pseudo-labels.

**Naive Source-Guided (Naive SG) versions.** To implement the Naive Source-Guided UDAP and MMT, resp. UDAP +Naive SG and MMT + Naive SG, the Source-Guided Pseudo-Labeling algorithm in Alg. 1 is directly applied to these methods.

**Framework implementation details.**

UDAP and MMT split their training into 2 phases:

- The initialization phase, where the model is only trained with the labeled source data.
- The Pseudo-Labeling phase, where the model use the pseudo-labeled target samples for training.

**Initialization phase.** We follow the guidelines and implementation details for supervised training from the paper [92] adopted by MMT [41]. Random Erasing Data augmentation [177] is not used during the initialization phase since it may reduce transferability of the source model features to the target domain thus generating more errors in pseudo-labels after UDA initialization [92].

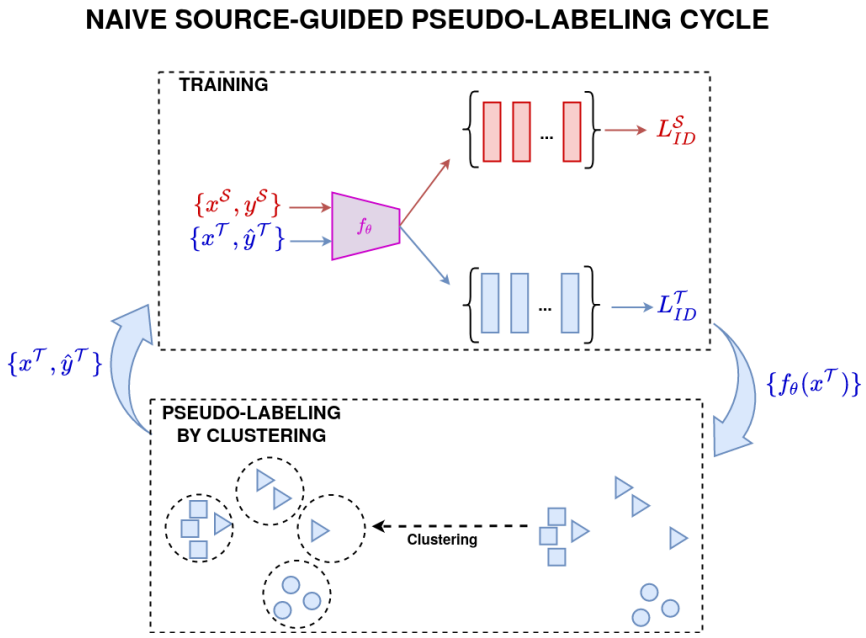


Figure 3.2: Illustration of the Naive Source-Guided Pseudo-Labeling cycle. An initial feature encoder $f_\theta$ trained on the source data $\{x^{\mathcal{S}}, y^{\mathcal{S}}\}$, is used to extract features $\{f_\theta(x^{\mathcal{T}})\}$ from the target training set images $\{x^{\mathcal{T}}\}$, which are used to predict pseudo-labels by clustering $\{x^{\mathcal{T}}, \hat{y}^{\mathcal{T}}\}$. $f_\theta$ is then fine-tuned by feature re-ID learning by minimizing the Source-Guided re-ID loss function $\mathrm{L_{ID}}$ (Eq. 3.1) composed of $\mathrm{L}^{\mathcal{S}}_{\mathrm{ID}}$ computed with the labeled source data and $\mathrm{L}^{\mathcal{T}}_{\mathrm{ID}}$ computed with the pseudo-labeled target data. Pseudo-Labeling methods perform several Pseudo-Labeling cycles to update and improve the pseudo-labels, which subsequently improves the performance of the learned feature encoder.

**Pseudo-labeling phase.** We use the same initialization phase preprocessing with two batches of 64 images, adding Random Erasing Data augmentation [177]: one for source images and another one for target. We feed separately the network with the source and target batches to ensure domain-specific batch normalization statistics. Other hyperparameters (clustering algorithm parameters, triplet loss margin, number of iterations for pseudo-labeling,...) used after the initialization phase are kept the same as the UDA paper's ones (resp MMT's ones): they correspond to the best hyperparameters found after validation on the target test set in their papers.

**Cross-dataset benchmarks.** The frameworks are evaluated, as well as their Naive SG version, on the commonly tested Duke→Market (Duke being the labeled source and Market the unlabeled target dataset) and analogously Market→Duke UDA tasks. Besides, the more challenging adaptation tasks Market→MSMT and Duke→MSMT UDA tasks are considered. Mean average precision (mAP) and CMC

rank-1 accuracy are reported to measure SG Pseudo-Labeling's performance on the target domain. The re-ID dataset details have been given in Sec. 2.2.3.

Table 3.1: Comparison of original target-only baselines with their corresponding Naive SG version for different person cross-dataset benchmarks. mAP and rank-1 are reported in %.

| Method | Duke→Market | | Market→Duke | | Duke→MSMT | | Market→MSMT | |
|---|---|---|---|---|---|---|---|---|
| | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 |
| UDAP ([120]) | **54.3** | **73.5** | **50.1** | **70.1** | **14.8** | **36.1** | **11.6** | **29.8** |
| UDAP ([120]) + Naive SG | 43.1 | 60.8 | 36.7 | 57.4 | 9.4 | 31.1 | 6.9 | 28.9 |
| MMT ([41]) | **71.2** | **87.7** | **65.1** | **78.0** | **23.5** | **50.0** | **22.9** | **49.2** |
| MMT ([41]) + Naive SG | 59.7 | 80.2 | 45.5 | 69.3 | 17.9 | 43.0 | 16.8 | 37.1 |

**Results and discussion.** As shown in Tab. 3.1, the Naive SG Pseudo-Labeling significantly degrades the cross-domain performance of every state-of-the-art Pseudo-Labeling method it is implemented on, and for all cross-dataset tasks. The performance drop is so significant that we can suppose that the proposed Naive SG model is probably too simple to leverage the source data, in addition to the target data, without harming the cross-domain performance. Indeed, we can assume that the source domain might bias the learning process. In their work on the classification task, Ben David et al. [8] estimate $\alpha$ in $L_{ID}$ (Eq. 3.1) so that the cross-domain performance cannot be harmed by the source data (in that case the estimated $\alpha$ should be equal to 0). In our case, for the UDA re-ID task, $\alpha$ has to be set as an hyperparameter since no target ground-truth label is available. To better address this aspect, we propose another Source-Guided Pseudo-Labeling model that can leverage the source data in order to consistently improve the cross-domain performance, even with an arbitrarily set value for $\alpha$.

### 3.2.2 Avoiding the source-bias in Source-Guided Pseudo-Labeling UDA re-ID

#### Two-head architecture

To avoid biasing the model and thus the discriminativeness of the target re-ID features with the source data, a Two-head neural network architecture for the feature encoders for $f_\theta$ as illustrated on Fig 3.3. It is composed of a common domain-

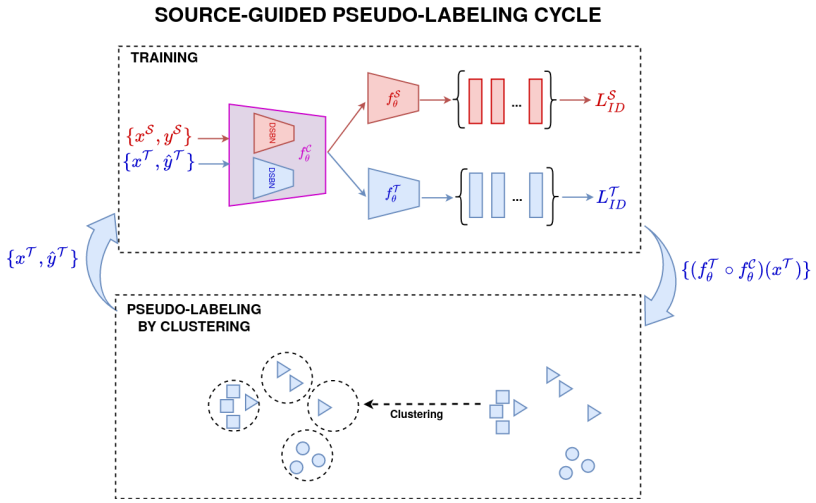


Figure 3.3: Illustration of the Source-Guided Pseudo-Labeling cycle. An initial feature encoder $f_\theta^{\mathcal{S}} \circ f_\theta^{\mathrm{C}}$ trained on the source data $\{x^{\mathcal{S}}, y^{\mathcal{S}}\}$, is used to initialize a Two-head feature encoder $f_\theta$ such that $f_\theta^{\mathcal{S}} \circ f_\theta^{\mathrm{C}} = f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}$. Target features are extracted $\{f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}(x^{\mathcal{T}})\}$ from the target training set images $\{x^{\mathcal{T}}\}$, which are used to predict pseudo-labels by clustering $\{x^{\mathcal{T}}, \hat{y}^{\mathcal{T}}\}$. $f_\theta$ is then fine-tuned by feature re-ID learning by minimizing the Source-Guided re-ID loss function $\mathrm{L_{ID}}$ (Eq. 3.1) composed of $\mathrm{L_{ID}^{\mathcal{S}}}$ computed with the labeled source data and $\mathrm{L_{ID}^{\mathcal{T}}}$ computed with the pseudo-labeled target data. Domain-specific batch normalization (DSBN) is used during training and feature extraction. Pseudo-Labeling methods perform several Pseudo-Labeling cycles to update and improve the pseudo-labels, which subsequently improves the performance of the learned feature encoder.

shared encoder $f_\theta^{\mathrm{C}}$ that computes low and mid-level features and two domain-specific encoders $f_\theta^{\mathcal{S}}$ and $f_\theta^{\mathcal{T}}$ that compute resp. the source and target high-level features. This choice of modeling is supported by work suggesting that features specialize for tasks in the top layers of the network [149]. Supposing that task-specialization, in our case, biases the learned representation toward a domain, this Two-head architecture is expected to prevent a source-specialization (bias) that could degrade the target feature representation (high-level features). With this Two-head architecture, the source data should directly benefit $f_\theta^{\mathrm{C}}$, where the specialization (bias) is lower, without degrading $f_\theta^{\mathrm{T}}$ where the domain-specialization is higher. Therefore, by improving $f_\theta^{\mathrm{C}}$, the target feature encoder $f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}$ should also be improved. Therefore, $f_\theta$ can be split into two separate domain-specific feature encoder given by $f_\theta^{\mathcal{S}} \circ f_\theta^{\mathrm{C}}$ for to compute the source features and $f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}$ for the target ones.

**Domain-specific batch normalization (DSBN)**

Common neural network architectures for re-ID contain batch normalization to improve the training convergence. Experiments from the paper [154] suggest that domain shift in data can reduce performance if the statistics of batch normalization layers are not computed separately for each domain. Since data from two different domains are used for learning, the Domain-specific batch normalization follows the paper suggestion and therefore computes statistics separately for source and target data.

The Source-Guided Pseudo-Labeling algorithm is given by the pseudo-code in Alg. 2. The general Source-Guided Pseudo-Labeling cycle is also illustrated on Fig. 3.3.

**Algorithm 2** Source-Guided Pseudo-Labeling

**Input:** Labeled source data $\mathrm{D}^{\mathcal{S}} = (\mathrm{X}^{\mathcal{S}}, \mathrm{Y}^{\mathcal{S}})$, unlabeled target data $\mathrm{X}^{\mathcal{T}}$, clustering algorithm C, an initialization phase re-ID loss function $\mathrm{L}_0^{\mathcal{S}}$, a Source-Guided Pseudo-Labeling phase re-ID loss function $\mathrm{L}_{\mathrm{ID}}$, a number of Pseudo-Labeling iterations $\mathrm{N}_{iter}$, an initial encoder $f_\theta^{(0)}$

1: Train the initial encoder $f_\theta^{(0)}$ on $\mathrm{D}^{\mathcal{S}}$ by optimizing $\mathrm{L}_0^{\mathcal{S}}$
2: Initialize **Two-head** $f_\theta$ such that $\mathbf{f}_\theta^{\mathcal{S}} \circ \mathbf{f}_\theta^{\mathbf{C}} = \mathbf{f}_\theta^{(\mathbf{0})}$ and $\mathbf{f}_\theta^{\mathcal{T}} \circ \mathbf{f}_\theta^{\mathbf{C}} = \mathbf{f}_\theta^{(\mathbf{0})}$ with **domain-specific bath normalization** layers.
3: **for** $t = 1$ to $\mathrm{N}_{iter}$ **do**
4: Compute target features: $\mathrm{F}^{\mathcal{T}} \leftarrow f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}(\mathrm{X}^{\mathcal{T}})$
5: Pseudo-label the target samples by clustering: $(\mathrm{X}^{\mathcal{T}}, \hat{\mathrm{Y}^{\mathcal{T}}}) \leftarrow \mathrm{C}(\mathrm{F}^{\mathcal{T}}, \mathrm{X}^{\mathcal{T}})$
6: Train $f_\theta$ during $\mathrm{N}_{epoch}$ by minimizing $\mathrm{L}_{\mathrm{ID}}$ using $\hat{\mathrm{D}^{\mathcal{T}}} = (\mathrm{X}^{\mathcal{T}}, \hat{\mathrm{Y}^{\mathcal{T}}})$ and $\mathrm{D}^{\mathcal{S}}$
7: **end for**
8: Return $f_\theta^{\mathcal{T}} \circ f_\theta^{\mathrm{C}}$

**Experiment details.**

**Two-head architecture.** Unless otherwise specified, we use as the common feature encoder all but the layers from the last convolutional block and after (4 first blocs of layers) in the ResNet-50.

**Domain-specific batch normalization.** Domain-specific batch normalization is implemented by replacing each classical batch normalization layer, with two batch normalization layers: one is dedicated for the source data, and the other for the

target one. During the forward feature extraction, the batch normalization of the source data (resp. the target data) is therefore specifically performed by the source (resp. the target) batch normalization layers.

**Source-Guided Pseudo-Labeling performance.**

Table 3.2: Comparison of original target-only baselines with their corresponding SG version for different person cross-dataset benchmarks. mAP and rank-1 are reported in %.

| Method | Duke→Market | | Market→Duke | | Duke→MSMT | | Market→MSMT | |
|---|---|---|---|---|---|---|---|---|
| | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 |
| Direct Transfer | 19.1 | 61.9 | 11.9 | 46.0 | 9.4 | 27.0 | 7.1 | 19.4 |
| UDAP ([120]) | 54.3 | 73.5 | 50.1 | 70.1 | 14.8 | 36.1 | 11.6 | 29.8 |
| UDAP ([120]) + SG | **59.1** | **80.8** | **55.6** | **73.2** | **19.3** | **45.6** | **14.9** | **35.4** |
| MMT ([41]) | **71.2** | 87.7 | **65.1** | 78.0 | 23.5 | 50.0 | 22.9 | 49.2 |
| MMT ([41]) + SG | 70.5 | **88.1** | 64.8 | **78.5** | **27.5** | **56.1** | **23.5** | **50.2** |
| Supervised Learning | 84.4 | 93.5 | 68.8 | 82.9 | 50.2 | 76.3 | 50.2 | 76.3 |

In Table 3.2 the Source-Guided frameworks (+SG) are compared with their original target-only versions on different cross-dataset benchmarks.

On all cross-dataset tasks, UDAP+SG outperforms the original UDAP: +4.8 p.p. mAP on Duke→Market, +5.5 p.p. mAP on Market→Duke, +4.5 p.p. mAP on Duke→MSMT and +3.3 p.p. mAP on Market→MSMT. This suggests that the proposed SG Pseudo-Labeling method, designed to reduce the source-bias of the target feature, manages to take advantage of the source data to improve the cross-domain performance.

For MMT+SG, the performance remains more or less the same as the original MMT on Duke→Market and Market→Duke. However, on cross-dataset tasks with the challenging MSMT as target, the performance are significantly improved by MMT+SG compared to MMT: +4.0 p.p. mAP on Duke→MSMT and +0.6 p.p. mAP on Market→MSMT. We think that Source-Guided Pseudo-Labeling may help reduce the impact of errors in pseudo-labels on the cross-domain performance. Indeed, on Duke→Market and Market→Duke the Mutal Mean Teachning technique might be redundant with SG, which could explain why performance does not increase. On the challenging cross-dataset tasks Duke→MSMT and Market→MSMT, the percentage of errors in pseudo-labels is expected to be greater, and so SG can have a

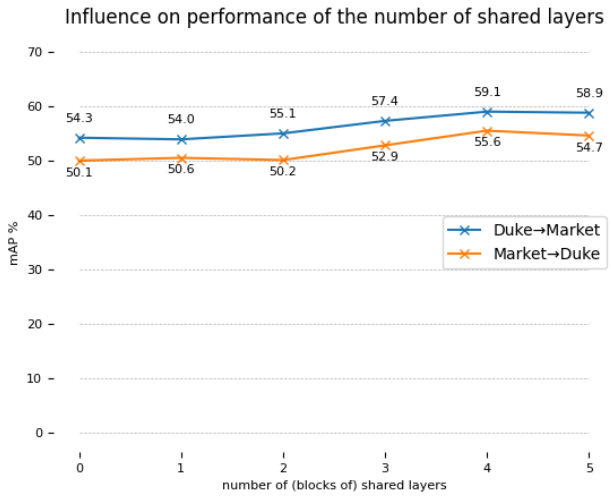


Figure 3.4: Impact on mAP (in %) of the number of shared layers used in the Two-head feature encoder $f_\theta^C$ of UDAP + SG, on Duke→Market and Market→Duke.

positive impact on the performance even if Mutual Mean Teaching is already used.

At the time of this work on Source-Guided Pseudo-Labeling, MMT being the best state-of-the-art method for UDA re-ID, SG further helps to helps further improve the cross-domain performance for challenging adaptation tasks with MSMT as target [30].

### 3.2.3 An ablative study on the bias robustness techniques.

This section aims at understanding how the different parts of the Source-Guided Pseudo-Labeling UDA approach improve the cross-domain re-ID performance.

### 3.2.4 Efficiency of the Two-head architecture.

**Goal.** As explained and motivated in Section 3.2.2, we propose a Two-head architecture to learn domain-specific high level ID discriminative features based on low and mid level domain-shared features learned with labeled source data and pseudo-labeled target data. We can wonder if our Two-head feature encoder manages to leverage the source samples to improve the target features. Furthermore, we would like to know how many layers should be shared to take advantage from the labeled source data without negatively biasing the target features.

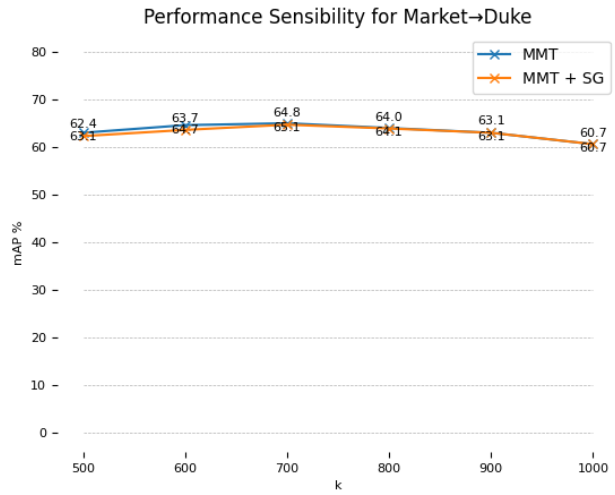


Figure 3.5: Robustness of MMT + SG to $k$ parameter's changes ($k$ controls k-means number of clusters) compared to the target-only framework MMT on Market→Duke.

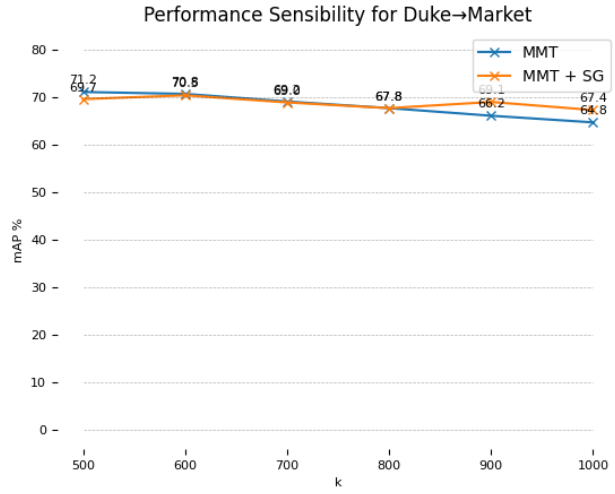


Figure 3.6: Robustness of our MMT + SG to $k$ parameter's changes ($k$ controls k-means number of clusters) compared to the target-only framework MMT on Duke→Market.

**Protocol.** To answer these two questions, we vary the number of ResNet-50 layers shared between source and target domains through the $f_\theta^C$ encoder of our Source-Guided UDAP. The ResNet-50 architecture can be divided into 5 convolutional blocks of layers defined in the ResNet paper [51] to which we refer to vary the number of shared layers. Case "0 shared layer" corresponds to the classical target-only Pseudo-Labeling methods (Fig. 5.2(1)) which corresponds to UDAP, where case "5

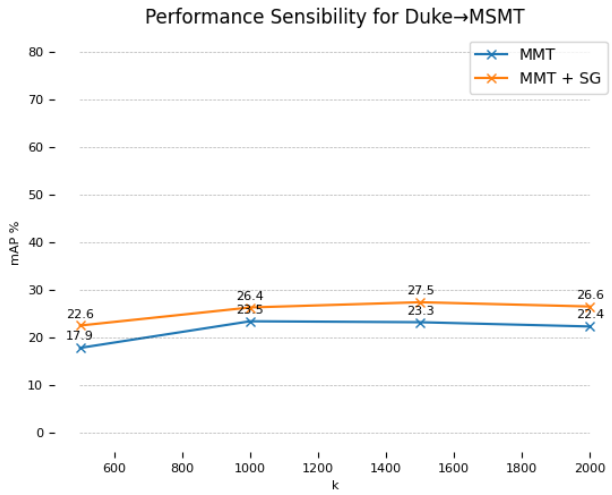


Figure 3.7: Robustness of MMT + SG to $k$ parameter's changes ($k$ controls k-means number of clusters) compared to the target-only framework MMT on Duke→MSMT.

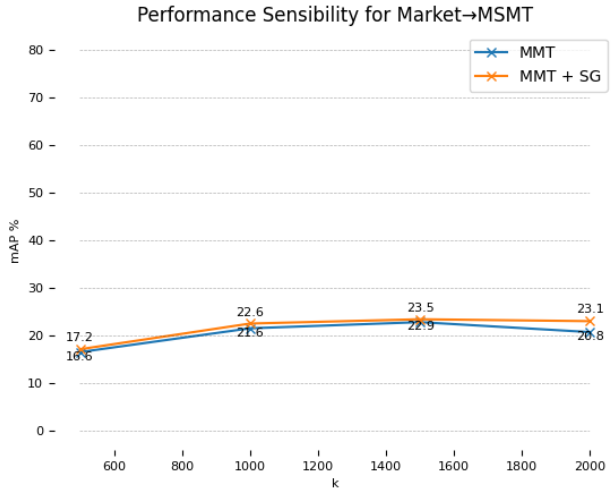


Figure 3.8: Robustness of MMT + SG to $k$ parameter's changes ($k$ controls k-means number of clusters) compared to the target-only framework MMT on Market→MSMT.

shared layers" to sharing the whole ResNet50 between source and target domains.

**Results.** Experiments show on Fig. 3.4 increasing performances when the number shared layers increase. More precisely, the best mAP is reached for 4 shared block of layers: 59.1% mAP for Duke→Market and 55.6% mAP for Market→Duke, increasing resp. the performances by 4.8 p.p. and 5.5 p.p. compared to the target only

Table 3.3: Impact of Domain-specific batch normalization on domain adaptation performance (mAP in %) .

| Methods | Market→Duke | Duke→Market |
|---|---|---|
| | mAP | mAP |
| UDAP | 50.1 | 54.3 |
| UDAP + SG w/o DSBN | 36.7 | 43.1 |
| UDAP + SG (w/ DSBN) | **55.6** | **59.1** |

model. The Source-Guided UDAP outperforms the classical target-only Pseudo-Labeling UDAP and our Two-head strategy gives the best results for Duke→Market and Market→Duke. This suggests that the proposed Two-head architecture in SG helps to improve the cross-domain performance.

### 3.2.5 Efficiency of Specific Batch Normalization

**Goal.** We study the effectiveness of domain-specific batch normalization as motivated in Sec. 3.2.2.

**Protocol.** To do so, we compare UDAP + SG (w/ DSBN) framework to a version that shares the batch normalization between domains as in Naive SG: UDAP + SG w/o DSBN.

**Results** In Tab. 3.3, we notice that sharing the batch normalization deteriorates the performance on both couples of Market→Duke and Duke→Market adaptation datasets. mAP drops more than 10 p.p. below the model using only the target data (UDAP). Only the addition of domain-specific batch normalization increases the performances of SG Pseudo-Labeling above the UDAP model. These experiments therefore show that the use of domain-specific batch normalization is an essential key of SG Pseudo-Labeling in order not to deteriorate the learning of discriminative target features by biasing the batch normalization statistics.

#### Is our strategy of using source samples robust to clustering parameters changes ?

In the UDA setting, choosing or tuning hyperparameter is a tricky task due to the absence of a labeled validation set for the target domain. It is therefore important in practice to design UDA methods robust to hyperparameter changes. In

particular, Pseudo-Labeling UDA methods [120] [41] give experimental evidences that performance can be very sensible to clustering parameters changes. That's why we would like to focus on the performance of our Source-Guided framework when these clustering parameters change. We choose to perform experiments using the Source-Guided MMT + SG, comparing it to its target-only version MMT, when the clustering parameter varies. We therefore change the $k$ parameter of k-means as in the MMT paper [41]. The $k$ parameter determines the number of clusters in the MMT frameworks. We choose the same interval of values as in the MMT paper [41] for varying the $k$ parameter.
For Market→Duke and Duke→Market in Fig. 3.5 and 3.6, the addition of the source term with MMT + SG does not seem to increase the maximum performance. However, there are quite different performance curve trends between MMT and MMT + SG: SG Pseudo-Labeling seems to be more robust for $k$ values above 800, i.e. when a number of clusters is chosen above the actual number of ground-truth identities. While MMT already proposes a strategy of resistance to pseudo-label noise, which can explain the non improvement of the best mAP, the addition of the source-guidance in MMT + SG seems to confer more stability to the clustering parameter variations. This stability conferred by the source is interesting given that we do not know the number of identities of the training set target, which can only be estimated at best.

In the more challenging cases where MSMT is the target dataset, there is a clear contribution from the source. We can see in Fig. 3.8 and 3.7 that it is stable to the change in $k$ and allowed to increase the maximum performance: from 23.5% to 27.5% for Duke→MSMT and from 22.9% to 23.5% for Market→MSMT. There is also a higher source contribution at high $k$ values. It can be assumed that MMT + SG works better in this more challenging case of adaptation because of the presence of more noisy labels during the transfer of the source model for initialization of Pseudo-Labeling: adding our strategy of exploiting source data therefore presents less redundancy with the one already implemented in the MMT framework, and even more if we "over-estimate" the number of clusters.

## 3.3 Conclusion and discussion

In line with existing work for classification, an empirical approach to exploit source data in addition to pseudo-labeled target data has been proposed: Source-

Guided Pseudo-Labeling UDA re-ID. This can be applied in general to any existing Pseudo-Labeling method, by symmetrically combining the two signals from the source and target data for model learning. It has been shown experimentally that simple use of the source can lead to performance degradation due to the difference in the data domains. To overcome this problem, as well as the source bias, two practices have been proposed: the Domain-specific batch normalization and the Two-head feature encoder. On several cross-dataset benchmarks, this allowed Source-Guided Pseudo-Labeling to improve the cross-domain performance of existing methods compared to using the target data alone. However, the proposed Source-Guided approach has some limitations. Firstly, the performance improvement by using the source is not consistent for all cross-datasets, and may even reduce them slightly. It would be ideal if Source-Guided Pseudo-Labeling could consistently improve cross-domain performance, or to know whether it can improve it or not. The empirical approach that led to the proposal of this Source-Guided method does not allow us to ensure this consistency, nor to determine the conditions that are beneficial to this improvement in performance by the source. The techniques proposed to avoid a negative impact of the use of source data are also part of this empirical approach, and are very specific. It is therefore not known whether there are other good practices to adopt in order to benefit from the source in a Pseudo-Labeling method. A theoretical approach to Pseudo-Labeling could answer these various limitations and questions raised.

# Chapter 4

# A formal approach to good practices in Pseudo-Labeling for UDA re-ID

## 4.1 Introduction

This chapter follows on from the previous one, having shown empirically that the source data can improve the performance of re-ID in cross-domain, through pseudo-labeling UDA methods. Using the source data during pseudo-labeling training therefore can be a general good practices, which can improve the performance of any pseudo-labeling UDA re-ID method. But this performance improvement is not consistent and seems subject to certain conditions. What are these conditions ? Are there other general good practices for pseudo-labeling UDA related or not to the use of the source ? This chapter aims at further exploring this direction by developing a new theoretical framework for Source-Guided Pseudo-Labeling UDA re-ID. To this end, after describing our motivation in Sec. 4.2, this chapter proposes three contributions that can be summarized as follows:

1) A novel theoretical framework for Pseudo-Labeling UDA re-ID, formalized through a new general learning upper-bound on the UDA re-ID performance (in Sec. 4.3).

2) General good practices for Pseudo-Labeling, directly deduced from the interpretation of the proposed theoretical framework, in order to improve the target re-ID performance, as well as possible implementations of these practices (in Sec. 4.3).

3) Extensive experiments on challenging person and vehicle cross-dataset re-ID tasks, showing consistent performance improvements for various state-of-the-art methods and various proposed implementations of good practices (in Sec. 4.6).

In Sec. 4.7, we conclude this chapter and discuss another aspect of pseudo-labeling outside the scope of the theoretical framework developed in this chapter.

This chapter was submitted as a paper to CVIU journal and is under review (preprint version [29]).

## 4.2 Motivations

### 4.2.1 A lack of theoretical work on Pseudo-Labeling UDA

Recent UDA re-ID approaches widely rely on the use of *pseudo-labels* for the target domain data ([120, 159, 62, 124, 156, 183, 143, 14, 41, 157, 163, 183, 103, 160, 144]). In fact, learning with these pseudo-labeled target samples can lead to a better identity-discriminative representation on the target domain. For this purpose, researchers designed a wide range of Pseudo-Labeling frameworks, based on a wide variety of modeling choices. Therefore, various directions have been explored to improve Pseudo-Labeling UDA re-ID approaches: some works focus on improving the predicted pseudo-labels ([61, 156, 124, 183, 16, 167, 175, 160]), others on reducing the impact of pseudo-label errors during training ([41, 163, 157, 62, 87, 38, 142, 80, 144, 80, 144, 35, 42]). These Pseudo-Labeling practices, integrated in specific Pseudo-Labeling methods, have been shown experimentally to improve the performance. However, even if these directions are considered as good practices for pseudo-labeling, there is no general work to highlight the conditions for their effectiveness outside of the specific frameworks they are implemented in. Moreover, it is still unknown how they improve the performance and how these techniques interact between each others to improve the performance. Moreover, some of these practices are not unanimously agreed among Pseudo-Labeling UDA re-ID work: the most outstanding example is leveraging ([30, 42, 59, 31]) or alternatively discarding ([159, 156]) the available ground-truth labeled source data to optimize the model jointly with the pseudo-labeled target data. In fact, the majority of Pseudo-Labeling UDA re-ID approaches do not use the available source

training set ([159, 156, 120, 62, 124, 156, 183, 143, 14]) after having access to target pseudo-labels. However, work conducted in the previous chapter, shows experimentally that the source data, continuously used in addition to the pseudo-labeled target data, can improve re-ID performance on the target domain. Other works, including the best state-of-the-art methods ([30, 42, 31, 59]) also leverage the source data during training. Therefore it can be asked whether the source data help to improve the cross-dataset re-ID performance of any Pseudo-Labeling approach. Or is it specific to these empirical method designs, using the source under the right conditions ? If so, are there some conditions under which any Pseudo-Labeling UDA re-ID method can benefit from the source data, after having access to target pseudo-labels ? These questions are left unanswered and may limit the performance of some target-only pseudo-labeling methods by discarding the source data, or degrade the performance of source-guided methods that do not use it correctly. We reckon this is due to the lack of theoretical work for pseudo-labeling UDA re-ID.

### 4.2.2 A lack of theoretical framework for source-guided pseudo-labeling UDA re-ID

To our knowledge, UDAP ([120]) is the only work which offers a theoretical framework on Pseudo-Labeling UDA re-ID. However, the theoretical framework proposed in this chapter differs from UDAP ([120]) on various aspects. First, our theoretical framework models directly the errors in the pseudo-labels contrary to UDAP ([120]), that focuses on how the target clusters are distributed in the feature space. Moreover, we propose a Pseudo-Labeling learning upper-bound directly on the target re-ID performance, while UDAP ([120]) focuses on a measure of clusterability on the target domain. Therefore, UDAP ([120]) does not model the impact of errors in pseudo-labels during the training. In other words, the theoretical framework of UDAP ([120]) aims at justifying their Pseudo-Labeling self-learning paradigm by focusing more specifically on clusterability of the target feature space, while ours seeks to encompass a majority of researches and good practices that improve empirically the UDA re-ID performance.

### 4.2.3 Theoretical works on Pseudo-Labeling UDA classification cannot be applied

It is relevant to ask whether theoretical work exists for Pseudo-Labeling UDA classification, a more investigated task than re-identification in the machine learning field. However, to our knowledge, there is no work that jointly models the impact of using source data in addition to pseudo-labeled target data. One of the pioneering works on Pseudo-Labeling for UDA ([116]), considers in its theoretical developments the use of pseudo-labels without the source data to tackle the case of "non-conservative" domain adaptation. Their theoretical framework does not model the impact of errors in pseudo-labels on the classification accuracy nor the use of the source samples after pseudo-label generation. The closest theoretical work would be that of Ben David et al. ([8]), in which they consider the problem of finding a model that minimizes the risk on the target domain, by minimizing an empirical risk jointly with labeled source and labeled target data. Nevertheless, this Source-guided theoretical work is not completely applicable in our case, as it considers the use of ground-truth target labels rather than pseudo-labels. From this point of view, the work of Natarajan et al. ([98]) models the impact of errors in the labels on the classification accuracy. However, the use of labeled data from another domain is not taken into account and therefore it is incomplete to model a domain adaptation problem.
Our work proposes a Source-guided Pseudo-Labeling theoretical framework bridging the work of Ben David et al. ([8]) and Natarajan et al. ([98]), specifically thought for UDA re-ID, in order to understand with a theoretical and general view all existing modeling practices in the UDA re-ID literature. Contrary to existing work, ours aims at deducing general good practices for UDA re-ID from theoretical analysis and interpretation.

## 4.3 A novel theoretical framework for Source-Guided Pseudo-Labeling UDA re-ID

To have a general and theoretical framework that tries to encompass the variety of Pseudo-Labeling UDA re-ID practices, we should model the use of the source labeled data in addition to the pseudo-labeled target data for training the re-ID model. To do so, we choose to establish a new learning upper-bound on the target

re-ID performance, measured by an expected risk on the target domain. We expect this upper-bound to highlight the use of the source data and the pseudo-labeled target data during the training.
The first step will be to model and define this target expected risk for the re-ID task in Sec. 4.3.1. Then, we will focus on how to measure the impact of using the source data on the target re-ID performance, by defining different Domain Discrepancy measures in Sec. 4.3.2. Then, we will model the use of pseudo-label for the target data in Sec. 4.3.3. All these preliminary modeling put together will be used to define the Source-Guided Pseudo-Labeling problem in Sec. 4.3.6. Sec. 4.3.5, introduces preliminary lemmas that will be used in Sec. 4.3.6 to establish the desired upper-bound on the target risk in the Source-Guided Pseudo-Labeling framework.

### 4.3.1 Definitions and notations for the re-ID problem

re-ID can be learned by binary classification of pairs of images, as having the same identity or not (i.e. a verification task). We choose to formulate the re-ID problem this way. Indeed, reformulating the re-ID problem as a verification problem allows us to model it as a closed-set classification task. It is therefore expected that this modeling will simplify theoretical development by allowing us to reuse some results already established in other works for binary classification ([8]). Consequently, we consider an input space describing pairs of images $\mathscr{X} \in \mathbb{R}^p \times \mathbb{R}^p$, $p \in \mathbb{N}$ and an output space $\{-1;1\}$ where "1" represents the label assigned to a pair of images with the same identity ("-1" otherwise). Therefore, in this chapter, $x \in \mathscr{X}$ will represent a pair of images (or a pair of image feature vectors).
To measure the re-ID performance, we need to define a loss function L. Usually, the binary classification task is evaluated by the "0-1" loss. However, as we would like to highlight afterward the influence of the loss bounds on the performance of our model, we choose a slight modification of this loss that we call the "0-M" loss. Contrary to the "0-1" loss, we expect with the "0-M" loss to explicitly highlight the loss bound $M > 0$ in the established learning bound. Indeed, the loss bound may give some insight for Pseudo-Labeling, whereas using the "0-1" loss, bounded by "1", might hide this relevant information in a neutral multiplicative interaction $\times 1$ with another term of the learning upper-bound. The "0-M" loss is defined by:

$$\forall y, y' \in \{-1;1\}, \mathrm{L}(y, y') = \frac{\mathrm{M}}{2}\|y - y'\| = \mathrm{M}\mathbb{1}_{y \neq y'}, \tag{4.1}$$

where $M > 0$ represents the loss bound since: $\forall y, y' \in \{-1;1\}, \|L(y,y')\| \leq M$.
Our work particularly focuses on the UDA re-ID task that we specify hereafter.

### 4.3.2 Measuring the Domain Discrepancy for Unsupervised Domain Adaptation

#### General definitions and notations for UDA

To model UDA, we consider two domains S and T, resp. called the source and target domains. These domains can be described as pairs of distributions $S = (\mathscr{D}^{\mathscr{S}}, f_{\mathscr{S}})$ and $T = (\mathscr{D}^{\mathscr{T}}, f_{\mathscr{T}})$, where $\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}$ are resp. the source and target domain marginal input distributions defined on $\mathscr{X}$, and $f_{\mathscr{S}} : \mathscr{X} \to \{-1;1\}$, $f_{\mathscr{T}} : \mathscr{X} \to \{-1;1\}$ represent resp. the source and target domain (ground-truth) labeling functions.
UDA aims at finding a hypothesis function (also called a classifier) $h \in \mathscr{H} \subseteq \{-1;1\}^{\mathscr{X}}$ which minimizes the target expected risk $\epsilon_{\mathscr{T}}(h, f_{\mathscr{T}})$ related to the loss function L: $\epsilon_{\mathscr{T}}(h, f_{\mathscr{T}}) = \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{T}}}[L(h(x), f_{\mathscr{T}}(x))]$. L being our defined "0-M" loss (Eq. 4.1), the target expected risk $\epsilon_{\mathscr{T}}(h, f_{\mathscr{T}})$ can more particularly be expressed as:

$$\begin{aligned} \forall h \in \mathscr{H}, \epsilon_{\mathscr{T}}(h, f_{\mathscr{T}}) &= \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{T}}}[L(h(x), f_{\mathscr{T}}(x))] \\ &= \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{T}}}[M\mathbb{1}_{h(x)\neq f_{\mathscr{T}}(x)}]. \end{aligned} \tag{4.2}$$

More generally, we also define $\forall h, h' \in \mathscr{H}, \epsilon_{\mathscr{T}}(h,h') = \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{T}}}[L(h(x), h'(x))]$ as well as the notation shortcut $\forall h \in \mathscr{H}, \epsilon_{\mathscr{T}}(h) = \epsilon_{\mathscr{T}}(h, f_{\mathscr{T}})$. In the same way, we define the source expected risk $\forall h \in \mathscr{H}, \epsilon_{\mathscr{S}}(h) = \epsilon_{\mathscr{S}}(h, f_{\mathscr{S}}) = \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{S}}}[L(h(x), f_{\mathscr{S}}(x))] = \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{S}}}[M\mathbb{1}_{h(x)\neq f_{\mathscr{S}}(x)}]$ and $\forall h, h' \in \mathscr{H}, \epsilon_{\mathscr{S}}(h,h') = \mathbb{E}_{x\sim\mathscr{D}^{\mathscr{S}}}[L(h(x), h'(x))]$.
Under the UDA setting, we want to minimize $\epsilon_{\mathscr{T}}$ given by Eq. 4.2 w.r.t $h \in \mathscr{H}$, and for this, we have access to a set of i.i.d labeled source samples and a set of i.i.d unlabeled target samples.

#### Measuring the domain gap

With the general definitions and notations for UDA, we can define two different measures of the gap between the source and target domains. By quantifying this domain gap, we wish to highlight in the learning bound the influence of using data from the source domain, to optimize the expected risk on the target domain.

The first quantity that will be used to measure the domain gap is the ideal joint error. To introduce the ideal joint error, we first define the *ideal joint hypothesis* which represents the classifier that performs the best simultaneously on both domains. It is defined as:

$$h^* = \underset{h \in \mathscr{H}}{\operatorname{argmin}} \epsilon_{\mathscr{S}}(h) + \epsilon_{\mathscr{T}}(h). \tag{4.3}$$

And then, we can define the *ideal joint error*.

**Definition** If $h^*$ represents the ideal joint hypothesis, the *ideal joint error* $\lambda$ is defined by:

$$\lambda = \epsilon_{\mathscr{S}}\left(h^*\right) + \epsilon_{\mathscr{T}}\left(h^*\right). \tag{4.4}$$

Intuitively, a large ideal joint error indicates that we cannot expect to find a hypothesis that performs well on the target and source domains. This implies that we cannot find a classifier that performs well on the target domain only by minimizing $\epsilon_{\mathscr{S}}$.

Moreover, we introduce another notion to measure the domain gap in the learning bound. We refer for this to existing work on domain adaptation for classification. Particularly, we choose the definition of the $\mathscr{H}\Delta\mathscr{H}$-distance given by Ben David et al. ([8]), as it is tailored for the binary classification.

**Definition** For any pair of distributions $\mathscr{P}$ and $\mathscr{Q}$ defined on $\mathscr{X}$, we define the $\mathscr{H}\Delta\mathscr{H}$-distance as:

$$\begin{aligned} d_{\mathscr{H}\Delta\mathscr{H}}(\mathscr{P},\mathscr{Q}) = 2 \sup_{h,h' \in \mathscr{H}} |&\Pr_{x\sim\mathscr{P}}\left[h(x) \neq h'(x)\right] \\ &- \Pr_{x\sim\mathscr{Q}}\left[h(x) \neq h'(x)\right]|, \end{aligned} \tag{4.5}$$

where $\Pr_{x\sim\mathscr{P}}\left[h(x) \neq h'(x)\right]$ (resp. $\Pr_{x\sim\mathscr{Q}}\left[h(x) \neq h'(x)\right]$) denotes the probability of "$h(x) \neq h'(x)$" when $x \sim \mathscr{P}$ (resp. $x \sim \mathscr{Q}$). The $\mathscr{H}\Delta\mathscr{H}$-distance can be linked to the source and target expected risks with the following lemma:

**Lemma 1 (L1)** For any hypotheses $h, h' \in \mathscr{H}$,

$$\left|\epsilon_{\mathscr{S}}\left(h,h'\right) - \epsilon_{\mathscr{T}}\left(h,h'\right)\right| \leq \frac{\mathrm{M}}{2} d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}},\mathscr{D}^{\mathscr{T}}\right), \tag{4.6}$$

where $\mathrm{M} > 0$ is the "0-M" loss bound defined in Eq. 4.1.

*Proof.* Let $h, h' \in \mathscr{H}$. We can highlight the "0-M" loss bound by multiplying and

dividing by $\mathrm{M} > 0$ the $\mathcal{H}\Delta\mathcal{H}$ -distance expression given by definition in Eq. 4.5:

$$\begin{aligned} d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}},\mathcal{D}^{\mathcal{T}}\right) &= 2 \sup_{h,h'\in\mathcal{H}} \left|\Pr_{x\sim\mathcal{D}^{\mathcal{S}}}\left[h(x)\neq h'(x)\right] - \Pr_{x\sim\mathcal{D}^{\mathcal{T}}}\left[h(x)\neq h'(x)\right]\right| \\ &= \frac{2}{\mathrm{M}} \sup_{h,h'\in\mathcal{H}} \left|\mathrm{M}\Pr_{x\sim\mathcal{D}^{\mathcal{S}}}\left[h(x)\neq h'(x)\right] - \mathrm{M}\Pr_{x\sim\mathcal{D}^{\mathcal{T}}}\left[h(x)\neq h'(x)\right]\right|. \end{aligned} \tag{4.7}$$

The expectation of the indicator function for an event is the probability of that event. Then, by rewriting the probabilities in terms of expectations, we recover the expression of the expected risks given by Eq. 4.2:

$$\begin{aligned} d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}},\mathcal{D}^{\mathcal{T}}\right) &= \frac{2}{\mathrm{M}} \sup_{h,h'\in\mathcal{H}} \left| \mathop{\mathbb{E}}_{x\sim\mathcal{D}^{\mathcal{S}}}\left[\mathrm{M}\mathbb{1}_{h(x)\neq h'(x)}\right] - \mathop{\mathbb{E}}_{x\sim\mathcal{D}^{\mathcal{T}}}\left[\mathrm{M}\mathbb{1}_{h(x)\neq h'(x)}\right] \right| \\ &= \frac{2}{\mathrm{M}} \sup_{h,h'\in\mathcal{H}} \left|\epsilon_{\mathcal{S}}\left(h,h'\right) - \epsilon_{\mathcal{T}}\left(h,h'\right)\right|. \end{aligned} \tag{4.8}$$

By using the definition of the sup operator:

$$d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}},\mathcal{D}^{\mathcal{T}}\right) \geq \frac{2}{\mathrm{M}} \left|\epsilon_{\mathcal{S}}\left(h,h'\right) - \epsilon_{\mathcal{T}}\left(h,h'\right)\right|. \tag{4.9}$$

Which is equivalent to the following inequality since $\mathrm{M} > 0$:

$$\left|\epsilon_{\mathcal{S}}\left(h,h'\right) - \epsilon_{\mathcal{T}}\left(h,h'\right)\right| \leq \frac{\mathrm{M}}{2} d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}},\mathcal{D}^{\mathcal{T}}\right). \tag{4.10}$$

□

In this section, we have defined the UDA framework, and derive, from our definitions, Lemma 1 establish the desired learning upper-bound in will be useful afterward (to derive the Lemma 2 given by Eq. 4.17). As our work more particularly focuses on Pseudo-Labeling UDA approach, we set a specific framework for noisy labels to deal with this kind of UDA approaches in the following section.

### 4.3.3 Modeling Pseudo-Labeling with the noisy-label framework

As motivated in Sec. 2.3, we recall that this chapter focuses on the pseudo-label paradigm for UDA re-ID, i.e. learning with pseudo-labels on the target domain, since they performed the best among the UDA re-ID approaches. For this, it is necessary to choose a theoretical model in order to account for the use of pseudo-labels

in the learning bounds. To this end, we propose to adapt the model of learning with noisy labels. Indeed, the Pseudo-Labeling can be seen as using a strategy in order to obtain "artificial" labels for the unlabeled target data. Concretely, the Pseudo-Labeling process corrupts the unkwown labels of our target samples to give pseudo-labels used as supervision during training. Therefore, the pseudo-labeled target samples can be seen as samples from a corrupted target distributions $\tilde{\mathcal{T}} = (\mathcal{D}^{\mathcal{T}}, \tilde{f}_{\mathcal{T}})$ where $\tilde{f}_{\mathcal{T}}$ is a (pseudo-)labeling function $\tilde{f}_{\mathcal{T}} : \mathcal{X} \to \{-1; 1\}$, which can be non-deterministic.

Our goal is to highlight the influence of the pseudo-labels noise on the target performance. Therefore, we follow the noise model used by Natarajan et al. ([98]) to derive an upper-bound on a classification expected risk: the class-conditional random noise model. Following the class-conditional random noise model, we have:

$$\forall x \in \mathcal{X}, \begin{cases} \rho_{-1} = \Pr(\tilde{f}_{\mathcal{T}}(x) = 1 | f_{\mathcal{T}}(x) = -1) \\ \rho_{+1} = \Pr(\tilde{f}_{\mathcal{T}}(x) = -1 | f_{\mathcal{T}}(x) = 1). \end{cases} \tag{4.11}$$

with $\rho_{-1} + \rho_{+1} < 1$. In other words, the corruption process is independent on the sample and only depends on the class. Therefore, it can be described by $\rho_{-1}$ which represents the probability that a "-1" labeled sample is pseudo-labeled "1" by $\tilde{f}_{\mathcal{T}}$ and $\rho_{+1}$ the probability that a "1" labeled sample becomes "-1" after Pseudo-Labeling by $\tilde{f}_{\mathcal{T}}$.

In the presence of noise in the annotations, i.e. when using pseudo-labeled target samples, the target empirical risk $\hat{\epsilon}_{\mathcal{T}}$ associated to $\epsilon_{\mathcal{T}}$ becomes a biased estimate of $\epsilon_{\mathcal{T}}$, as shown in the work of Natarajan et al. ([98]). That's why, following their Lemma 1 ([98]), we define the *corrected loss function* $\tilde{\mathrm{L}}$:

$$\forall y, y' \in \{-1; 1\}, \tilde{\mathrm{L}}(y, y') = \frac{(1 - \rho_{-y'})\mathrm{L}(y, y') - \rho_{y'}\mathrm{L}(y, -y')}{1 - \rho_{+1} - \rho_{-1}}. \tag{4.12}$$

where $\rho_{y'} = \rho_{-1}$ if $y' = -1$ and $\rho_{y'} = \rho_{+1}$ if $y' = 1$. Therefore, also according to their Lemma 1 and following the previous notation:

$$\forall h \in \mathcal{H}, \mathbb{E}_{x \sim \mathcal{D}^{\mathcal{T}}}[\tilde{\mathrm{L}}(h(x), \tilde{f}_{\mathcal{T}}(x))] = \epsilon_{\mathcal{T}}(h). \tag{4.13}$$

In order to define the Source-guided empirical risk with target pseudo-labels in the

next section, we denote by $\tilde{\epsilon}_{\mathcal{T}}$ the *target empirical corrected risk* associated to the corrected loss function defined in Eq. 4.12, and computed with pseudo-labeled target samples. It is an unbiased estimate of $\epsilon_{\mathcal{T}}$ according to Eq. 4.13.

### 4.3.4 Establishing a new Learning Bound for Pseudo-Labeling UDA re-ID

To integrate the source in the training stage, we assume that we optimize a convex weighting of the source empirical risk associated to $\epsilon_{\mathcal{S}}$ and corrected target empirical risks, that we call the *Source-guided empirical risk with target pseudo-labels* $\hat{\epsilon}_\alpha$. For this purpose, we have $m$ training samples in total, of which $\beta m$ are i.i.d pseudo-labeled target samples $(x_i^{\mathrm{T}}, \tilde{y}_i^{\mathrm{T}})_{1\leq i\leq \beta m}$ from $\tilde{\mathcal{T}}$ and $(1-\beta)m$ are labeled source samples$(x_i^{\mathcal{S}}, y_i^{\mathcal{S}})_{1\leq i\leq (1-\beta)m}$ from S, $\beta \in [0,1]$. With these samples, $\hat{\epsilon}_{\mathcal{S}}$, $\tilde{\epsilon}_{\mathcal{T}}$ can be expressed as, for any $h \in \mathcal{H}$:

$$\begin{cases} \tilde{\epsilon}_{\mathcal{T}}(h) & = \frac{1}{\beta m}\sum_{i=1}^{\beta m} \tilde{\mathrm{L}}(h(x_i^{\mathrm{T}}), \tilde{y}_i^{\mathrm{T}}) \\ & = \frac{1}{\beta m}\sum_{i=1}^{\beta m} \tilde{\mathrm{L}}(h(x_i^{\mathrm{T}}), \tilde{f}_{\mathcal{T}}(x_i^{\mathrm{T}})) \\ \hat{\epsilon}_{\mathcal{S}}(h) & = \frac{1}{(1-\beta)m}\sum_{i=1}^{(1-\beta)m} \mathrm{L}(h(x_i^{\mathcal{S}}), y_i^{\mathcal{S}}) \\ & = \frac{1}{(1-\beta)m}\sum_{i=1}^{(1-\beta)m} \mathrm{L}(h(x_i^{\mathcal{S}}), f_{\mathcal{S}}(x_i^{\mathcal{S}})). \end{cases} \tag{4.14}$$

And then $\hat{\epsilon}_\alpha$ can be expressed as:

$$\hat{\epsilon}_\alpha = \alpha\tilde{\epsilon}_{\mathcal{T}} + (1-\alpha)\hat{\epsilon}_{\mathcal{S}}, \alpha \in [0,1]. \tag{4.15}$$

We also define the quantity $\epsilon_\alpha$ given by:

$$\epsilon_\alpha = \alpha\epsilon_{\mathcal{T}} + (1-\alpha)\epsilon_{\mathcal{S}}, \alpha \in [0,1]. \tag{4.16}$$

Therefore, $\hat{h} = \mathrm{argmin}_{h\in\mathcal{H}}\hat{\epsilon}_\alpha(h)$ will be our model learned by Source-guided Pseudo-Labeling, i.e., by minimizing the Source-guided empirical risk with target pseudo-labels $\hat{\epsilon}_\alpha$. In the UDA setting, we want this model to have the best re-ID performance on the target domain, i.e., we want to reduce $\epsilon_{\mathcal{T}}(\hat{h})$. Therefore, we would like to establish an upper-bound on $\epsilon_{\mathcal{T}}(\hat{h})$. As Ben David et al. ([8]), to establish a learning bound on $\epsilon_{\mathcal{T}}(\hat{h})$, we proceed in three steps:

- Linking $\epsilon_\alpha$ to the target expected risk $\epsilon_{\mathcal{T}}$, with the following Lemma 2 (cf Sec. 4.3.5).

- Linking $\hat{\epsilon}_\alpha$ to $\epsilon_\alpha$ with the following Lemma 3 (cf Sec. 4.3.5).
- Using Lemma 2 and Lemma 3 to build the desired upper-bound on $\epsilon_{\mathcal{T}}(\hat{h})$ (cf Sec. 4.3.6).

### 4.3.5 Preliminary lemmas to establish the Source-guided Pseudo-Labeling upper-bound for UDA re-ID

We first link $\epsilon_\alpha$ to the target expected risk $\epsilon_{\mathcal{T}}$ (that we wish to minimize) with the following lemma.

**Lemma 2 (L2)** Let $h$ be a classifier in $\mathcal{H}$. Then

$$|\epsilon_\alpha(h) - \epsilon_{\mathcal{T}}(h)| \leq (1-\alpha)\left(\frac{M}{2} d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}}, \mathcal{D}^{\mathcal{T}}\right) + \lambda\right). \tag{4.17}$$

*Proof.* Let $h \in \mathcal{H}$ and $h^*$ be the ideal joint hypothesis defined in Eq. 4.3. By definition of $\epsilon_\alpha$ (Eq. 4.16), we can write:

$$\begin{aligned} |\epsilon_\alpha(h) - \epsilon_{\mathcal{T}}(h)| &= |\alpha\epsilon_{\mathcal{T}}(h) + (1-\alpha)\epsilon_{\mathcal{S}}(h) - \epsilon_{\mathcal{T}}(h)| \\ &= (1-\alpha)\,|\epsilon_{\mathcal{S}}(h) - \epsilon_{\mathcal{T}}(h)| \\ &= (1-\alpha)|\epsilon_{\mathcal{S}}(h) - \epsilon_{\mathcal{S}}(h,h^*) + \epsilon_{\mathcal{S}}(h,h^*) - \epsilon_{\mathcal{T}}(h,h^*) \\ &\quad + \epsilon_{\mathcal{T}}(h,h^*) - \epsilon_{\mathcal{T}}(h)|. \end{aligned} \tag{4.18}$$

Applying the triangular inequality property of $|.|$, we obtain:

$$\begin{aligned} |\epsilon_\alpha(h) - \epsilon_{\mathcal{T}}(h)| \leq (1-\alpha)[&\left|\epsilon_{\mathcal{S}}(h) - \epsilon_{\mathcal{S}}\left(h,h^*\right)\right| + |\epsilon_{\mathcal{S}}\left(h,h^*\right) \\ &- \epsilon_{\mathcal{T}}\left(h,h^*\right)| + \left|\epsilon_{\mathcal{T}}\left(h,h^*\right) - \epsilon_{\mathcal{T}}(h)\right|]. \end{aligned} \tag{4.19}$$

Using the triangular inequality property of $\epsilon_{\mathcal{S}}(.,.)$ and $\epsilon_{\mathcal{T}}(.,.)$ , the inequality becomes:

$$\begin{aligned} |\epsilon_\alpha(h) - \epsilon_{\mathcal{T}}(h)| \leq (1-\alpha)[&\epsilon_{\mathcal{S}}\left(h^*\right) + \left|\epsilon_{\mathcal{S}}\left(h,h^*\right) - \epsilon_{\mathcal{T}}\left(h,h^*\right)\right| \\ &+ \epsilon_{\mathcal{T}}\left(h^*\right)]. \end{aligned} \tag{4.20}$$

And then by applying Lemma 1 (Eq. 4.6) and by definition of $\lambda$ (Eq. 4.4):

$$|\epsilon_\alpha(h) - \epsilon_{\mathcal{T}}(h)| \leq (1-\alpha)\left(\frac{M}{2} d_{\mathcal{H}\Delta\mathcal{H}}\left(\mathcal{D}^{\mathcal{S}}, \mathcal{D}^{\mathcal{T}}\right) + \lambda\right). \tag{4.21}$$

□

We can also link the empirical risk $\hat{\epsilon}_\alpha$ (Eq. 4.15), to the expected risk $\epsilon_\alpha$ (Eq. 4.16), with the following lemma.

**Lemma 3 (L3)** For any $\mu > 0$:

$$\Pr\left[|\hat{\epsilon}_\alpha(h) - \epsilon_\alpha(h)| \geq \mu\right] \leq 2\exp\left(\frac{-2m\mu^2}{\frac{4M^2\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{M^2(1-\alpha)^2}{1-\beta}}\right), \tag{4.22}$$

As Ben David et al. for their theorem 3 [8], we refer to Anthony and Bartlett ([4]) for the detailed classical steps to derive the following VC-dimension upper-bound, using the inequality concentration from our Lemma 3. If $d$ is the VC-dimension of $\mathscr{H}$, with a probability $\delta$ ($0 \leq \delta \leq 1$) on the drawing of the samples, we have:

$$|\epsilon_\alpha(h) - \hat{\epsilon}_\alpha(h)| \leq 2\sqrt{\frac{4M^2\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{M^2(1-\alpha)^2}{1-\beta}}\sqrt{\frac{2d\log(2(m+1)) + 2\log\left(\frac{8}{\delta}\right)}{m}}. \tag{4.23}$$

Before giving the Lemma 3's proof, we recall the Hoeffding's inequality:

**Hoeffding's inequality**. If $X_1, \ldots, X_n$ are independent random variables with $a_i \leq X_i \leq b_i$ for all $i$, then for any $\mu > 0$,

$$\Pr[|\bar{X} - E[\bar{X}]| \geq \mu] \leq 2\exp\left(\frac{-2n^2\mu^2}{\sum_{i=1}^{n}\left(\text{range}(X_i)\right)^2}\right), \tag{4.24}$$

where $\bar{X} = (X_1 + \cdots + X_n)/n$ and $\forall 1 \leq i \leq n, \text{range}(X_i) = b_i - a_i$.

*Proof.* (L3). Let $h \in \mathscr{H}$ and $X_1, \ldots, X_{\beta m}$ be some random variables taking the values:

$$\frac{\alpha}{\beta}\tilde{L}(h(x_1), \tilde{f}_{\mathscr{T}}(x_1)), \ldots, \frac{\alpha}{\beta}\tilde{L}(h(x_{\beta m}), \tilde{f}_{\mathscr{T}}(x_{\beta m})), \tag{4.25}$$

for the random variables $x_1 \ldots x_{\beta m}$ associated to the generation of the $\beta m$ samples from the pseudo-labeled target domain $\tilde{\mathscr{T}}$. Similarly, let $X_{\beta m+1}, \ldots, X_m$ be some random variables taking the value:

$$\frac{1-\alpha}{1-\beta}L(h(x_{\beta m+1}), f_{\mathscr{S}}(x_{\beta m+1})), \ldots, \frac{1-\alpha}{1-\beta}L(h(x_m), f_{\mathscr{S}}(x_m)), \tag{4.26}$$

for the random variables $x_{\beta m+1}, \ldots, x_m$ associated to the generation of the $(1-\beta)m$ samples from the source domain S.

By definition of $\hat{\epsilon}_\alpha$ (Eq. 4.15), we can write:

$$\hat{\epsilon}_\alpha(h) = \alpha\tilde{\epsilon}_{\mathcal{T}}(h) + (1-\alpha)\hat{\epsilon}_{\mathcal{S}}(h). \tag{4.27}$$

And by definition of $\tilde{\epsilon}_{\mathcal{T}}$ and $\hat{\epsilon}_{\mathcal{S}}$ (Eq. 4.14), we have:

$$\begin{aligned}\hat{\epsilon}_\alpha(h) &= \alpha\frac{1}{\beta m}\sum_{i=1}^{\beta m}\tilde{\mathrm{L}}(h(x_i), \tilde{f}_{\mathcal{T}}(x_i)) + (1-\alpha)\frac{1}{(1-\beta)m}\sum_{i=\beta m+1}^{m}\mathrm{L}(h(x_i), f_{\mathcal{S}}(x_i)) \\ &= \frac{1}{m}\left(\sum_{i=1}^{\beta m}\frac{\alpha}{\beta}\tilde{\mathrm{L}}(h(x_i), \tilde{f}_{\mathcal{T}}(x_i)) + \sum_{i=\beta m+1}^{m}\frac{(1-\alpha)}{(1-\beta)}\mathrm{L}(h(x_i), f_{\mathcal{S}}(x_i))\right).\end{aligned} \tag{4.28}$$

Then we can write, by definition of $\mathrm{X}_1, \ldots, \mathrm{X}_m$:

$$\hat{\epsilon}_\alpha(h) = \frac{1}{m}\sum_{i=1}^{m}\mathrm{X}_i. \tag{4.29}$$

Using the linearity of expectations, we have:

$$\begin{aligned}\mathbb{E}_x[\hat{\epsilon}_\alpha(h)] = \frac{1}{m}(&\frac{\alpha}{\beta}\sum_{i=1}^{\beta m}\mathbb{E}_{x_i\sim\mathcal{D}^{\mathcal{T}}}[\tilde{\mathrm{L}}(h(x_i), \tilde{f}_{\mathcal{T}}(x_i))] \\ &+ \frac{(1-\alpha)}{(1-\beta)}\sum_{i=\beta m+1}^{m}\mathbb{E}_{x_i\sim\mathcal{D}^{\mathcal{S}}}[\mathrm{L}(h(x_i), f_{\mathcal{S}}(x_i))]).\end{aligned} \tag{4.30}$$

According to Eq. 4.2:

$$\begin{aligned}\mathbb{E}_x[\hat{\epsilon}_\alpha(h)] &= \frac{1}{m}\left(\frac{\alpha}{\beta}\beta m\epsilon_{\mathcal{T}}(h) + \frac{1-\alpha}{1-\beta}(1-\beta)m\epsilon_{\mathcal{S}}(h)\right) \\ &= \alpha\epsilon_{\mathcal{T}}(h) + (1-\alpha)\epsilon_{\mathcal{S}}(h).\end{aligned} \tag{4.31}$$

And then by definition of $\epsilon_\alpha$ (Eq. 4.16):

$$\mathbb{E}_x[\hat{\epsilon}_\alpha(h)] = \epsilon_\alpha(h). \tag{4.32}$$

Moreover, by definition of $\tilde{\mathrm{L}}$ (Eq. 4.12) and L (Eq. 4.1), we have:

$$\forall y, y' \in \{-1; 1\}, \begin{cases} \frac{-\mathrm{M}}{1-\rho_{+1}-\rho_{-1}} \leq \tilde{\mathrm{L}}(y, y') \leq \frac{\mathrm{M}}{1-\rho_{+1}-\rho_{-1}} \\ 0 \leq \mathrm{L}(y, y') \leq \mathrm{M}. \end{cases} \tag{4.33}$$

Therefore, we can say that $\mathrm{X}_1, \ldots, \mathrm{X}_{\beta m} \in [-\frac{\mathrm{M}\alpha}{\beta(1-\rho_{+1}-\rho_{-1})}, \frac{\mathrm{M}\alpha}{\beta(1-\rho_{+1}-\rho_{-1})}]$ and

$X_{\beta m+1}, \ldots, X_m \in [0, \frac{M(1-\alpha)}{(1-\beta)}]$. And then, we have:

$$\text{range}(X_i) = \begin{cases} \frac{2M\alpha}{\beta(1-\rho_{+1}-\rho_{-1})}, 1 \leq i \leq \beta m \\ \frac{M(1-\alpha)}{1-\beta}, \beta m + 1 \leq i \leq m. \end{cases} \tag{4.34}$$

According to Eq. 4.30, $\forall \mu > 0$:

$$\Pr\left[|\hat{\epsilon}_\alpha(h) - \epsilon_\alpha(h)| \geq \mu\right] = \Pr\left[|\hat{\epsilon}_\alpha(h) - \mathbb{E}_x[\hat{\epsilon}_\alpha(h)]| \geq \mu\right]. \tag{4.35}$$

Then by applying the Hoeffding's inequality (Eq. 4.24) to $\hat{\epsilon}_\alpha(h)$:

$$\begin{aligned} \Pr\left[|\hat{\epsilon}_\alpha(h) - \epsilon_\alpha(h)| \geq \mu\right] &\leq 2\exp\left(\frac{-2m^2\mu^2}{\sum_{i=1}^{m} \text{range}\,(X_i)^2}\right) \\ &\leq 2\exp\left(\frac{-2m^2\mu^2}{\sum_{i=1}^{\beta m} \text{range}\,(X_i)^2 + \sum_{i=\beta m+1}^{m} \text{range}\,(X_i)^2}\right). \end{aligned} \tag{4.36}$$

Then according to Eq. 4.34:

$$\begin{aligned} \Pr\left[|\hat{\epsilon}_\alpha(h) - \epsilon_\alpha(h)| \geq \mu\right] &\leq 2\exp\left(\frac{-2m^2\mu^2}{\beta m\left(\frac{2M\alpha}{\beta(1-\rho_{+1}-\rho_{-1})}\right)^2 + (1-\beta)m\left(\frac{M(1-\alpha)}{1-\beta}\right)^2}\right) \\ &\leq 2\exp\left(\frac{-2m\mu^2}{\frac{4M^2\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{M^2(1-\alpha)^2}{1-\beta}}\right). \end{aligned} \tag{4.37}$$

□

### 4.3.6 A new learning bound for Pseudo-Labeling UDA

Using the previous notation, we define the *ideal target hypothesis* $h^*_{\mathcal{T}} = \arg\min_{h \in \mathcal{H}} \epsilon_{\mathcal{T}}(h)$. The upper-bound on $\epsilon_{\mathcal{T}}(\hat{h})$ can be established.

**Theorem** With a probability $1-\delta$ $(0 \leq \delta \leq 1)$ on the drawing of the samples, we have:

$$\epsilon_{\mathscr{T}}(\hat{h}) \leq \epsilon_{\mathscr{T}}(h^*_{\mathscr{T}}) + 4\mathrm{M}\overbrace{\sqrt{\frac{4\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{1-\alpha^2}{1-\beta}}}^{\mathscr{N}}\overbrace{\sqrt{\frac{2d\log(2(m+1)) + 2\log(\frac{8}{\delta})}{m}}}^{\mathscr{C}} + \underbrace{2\,(1-\alpha)(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}) + \lambda)}_{\mathscr{DD}}. \tag{4.38}$$

$\mathscr{N}$, $\mathscr{C}$ and $\mathscr{DD}$ correspond to noteworthy terms that will be discussed in the next section (Sec. 4.4) to get insight from the upper-bound.

*Proof.* Following the previous notations, let $0 \leq \delta \leq 1$. We first use the Lemma 2 (Eq. 4.17) to bound $\epsilon_{\mathscr{T}}(\hat{h})$:

$$\epsilon_{\mathscr{T}}(\hat{h}) \leq \epsilon_{\alpha}(\hat{h}) + (1-\alpha)\left(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}\right) + \lambda\right). \tag{4.39}$$

Then by using the upper-bound on $\epsilon_{\alpha}(\hat{h})$ derived from Lemma 3 (Eq. 4.23), we have:

$$\begin{aligned}\epsilon_{\mathscr{T}}(\hat{h}) \leq & \hat{\epsilon}_{\alpha}(\hat{h}) + 2\sqrt{\frac{4\mathrm{M}^2\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{\mathrm{M}^2(1-\alpha)^2}{1-\beta}}\sqrt{\frac{2d\log(2(m+1)) + 2\log\left(\frac{8}{\delta}\right)}{m}} \\ & + (1-\alpha)\left(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}\right) + \lambda\right).\end{aligned} \tag{4.40}$$

Since $\hat{h} = \arg\min_{h\in\mathscr{H}} \hat{\epsilon}_{\alpha}(h)$, we have $\hat{\epsilon}_{\alpha}(\hat{h}) \leq \hat{\epsilon}_{\alpha}(h^*_{\mathscr{T}})$, and therefore:

$$\begin{aligned}\epsilon_{\mathscr{T}}(\hat{h}) \leq & \hat{\epsilon}_{\alpha}\left(h^*_{\mathscr{T}}\right) + 2\mathrm{M}\sqrt{\frac{4\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{(1-\alpha)^2}{1-\beta}}\sqrt{\frac{2d\log(2(m+1)) + 2\log\left(\frac{8}{\delta}\right)}{m}} \\ & + (1-\alpha)\left(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}\right) + \lambda\right).\end{aligned} \tag{4.41}$$

And then, by using the Lemma 3 (Eq. 4.23) to bound $\hat{\epsilon}_{\alpha}(h^*_{\mathscr{T}})$, we have:

$$\begin{aligned}\epsilon_{\mathscr{T}}(\hat{h}) \leq & \epsilon_{\alpha}\left(h^*_{\mathscr{T}}\right) + 4\mathrm{M}\sqrt{\frac{4\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{(1-\alpha)^2}{1-\beta}}\sqrt{\frac{2d\log(2(m+1)) + 2\log\left(\frac{8}{\delta}\right)}{m}} \\ & + (1-\alpha)\left(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}\right) + \lambda\right).\end{aligned} \tag{4.42}$$

Finally, by using the Lemma 2 (Eq. 4.17) to upper-bound $\epsilon_\alpha(h^*_{\mathscr{T}})$, we have:

$$\begin{aligned}\epsilon_{\mathscr{T}}(\hat{h}) \leq & \epsilon_{\mathscr{T}}\left(h^*_{\mathscr{T}}\right) + 4\mathrm{M}\sqrt{\frac{4\alpha^2}{\beta(1-\rho_{+1}-\rho_{-1})^2} + \frac{(1-\alpha)^2}{1-\beta}}\sqrt{\frac{2d\log(2(m+1)) + 2\log\left(\frac{8}{\delta}\right)}{m}} \\ & + 2(1-\alpha)\left(\frac{\mathrm{M}}{2}d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}}, \mathscr{D}^{\mathscr{T}}\right) + \lambda\right).\end{aligned} \tag{4.43}$$

□

To the best of our knowledge, this bound is novel, and takes into account the interactions between the source data and the pseudo-annotated target data. Even if the overall upper-bound establishment is inspired by Ben David et al. ([8]), we recall that theirs does not take into account the use of pseudo-labels for the target data nor the loss bound given by M. Similarly, Natarajan et al. ([98]) does not take into account the use of source data for learning. In the following Sec. 4.4, we propose an interpretation of this bound in order to get more insight on Source-guided Pseudo-Labeling learning. Then, we derive good practices from it.

## 4.4 Interpretation of the bound and derived good practices

Taking into account errors in the pseudo-labels, as well as the domain discrepancy between source and target, leads to a new learning upper-bound on Source-guided Pseudo-Labeling for UDA re-ID given by Eq. 4.38 in Sec. 4.3.6. This bound could be used to find the best hyperparameter $\alpha$ to weight the source-guidance in the Source-guided Pseudo-Labeling empirical risk $\hat{\epsilon_\alpha}$ (Eq. 4.15) i.e. to find an optimal solution $\alpha^*$ minimizing the upper-bound as a function of $\alpha$. This optimization has been solved by Ben David et al. ([8]) with their upper-bound for binary classification (with target ground-truth labels), to find the optimal mixing value between source and target in $\hat{\epsilon}_\alpha$. However, in the case of our bound, $\alpha^*$ would eventually depend on the noise probabilities $\rho_{-1}$ and $\rho_{+1}$. These values are not known in practice. Therefore, the interest of looking for such a solution $\alpha$ is limited from a practical perspective. However, it is still possible to get insight about Source-guided Pseudo-Labeling for UDA re-ID by analyzing the different terms of the bound: $\mathscr{N}$, $\mathscr{C}$ and $\mathscr{DD}$ resp. for the Noise term, the Complexity term, and the Domain Discrepancy term. Therefore, we consider $\alpha$ as a hyperparameter specified before training the

model. Then, we propose to analyze this bound by looking at the influence on the target performance of its key elements: the complexity term, the noise term and the domain discrepancy term. This analysis contributes to answering the question of how to better use the source data during training with pseudo-labels, and to deduce general good practices to follow.

### 4.4.1 Noise term: $\mathscr{N}$

The noise term $\mathscr{N}$ of our new learning bound for Pseudo-Labeling UDA re-ID (Eq. 4.38 in Sec. 4.3.6) involves the noise probabilities $\rho_{-1}$ and $\rho_{+1}$, as well as $\alpha$ which describes the weight put on the (noisy) target data in the Source-guided Pseudo-Labeling empirical risk $\hat{\epsilon}_\alpha$ (Eq. 4.15) minimized during training. Intuitively, $\mathscr{N}$ represents the impact of errors in the pseudo-labels on the re-ID performance: the higher $\rho_{-1}$ and $\rho_{+1}$, the higher $\mathscr{N}$. Then, increasing $\mathscr{N}$ increases the upper-bound, and therefore is more likely to degrade the target re-ID performance $\epsilon_{\mathscr{T}}(\hat{h})$.
While $\rho_{-1}$ and $\rho_{+1}$ are unknown, it is possible to reduce them in order to reduce $\mathscr{N}$. Indeed, these probabilities could be estimated by the proportions of mislabeled pairs of pseudo-labeled data. Even if we cannot directly compute these quantities without the ground-truth target labels, it is possible to reduce them. For example, they can be reduced with a pseudo-label refinement strategy or with a strategy of filtering out the outliers (mis-pseudo-labeled data) in the target training set.

### 4.4.2 Complexity term: $\mathscr{C}$

The complexity term $\mathscr{C}$ of our new learning bound for Pseudo-Labeling UDA re-ID (Eq. 4.38) involves the VC-dimension $d$ of $\mathscr{H}$ measuring the complexity of our selected set of models. Moreover, it also involves the total number $m$ of training data, which includes the source and target samples. Intuitively, the complexity term measures how well the class of hypothesis can memorize the training dataset given its number of samples. According to the bound, reducing it should also reduce the expected target risk $\epsilon_{\mathscr{T}}(\hat{h})$ of our learned model $\hat{h}$. To reduce it, two options are available. On the one hand, we could reduce the VC-dimension $d$ of the hypothesis class: this is what is classically done in machine learning using regularization on the learned model parameters such as weight decay.
The other option is to increase $m$, and for that we have to use as much data as possible during the training. $m$ can be increased artificially by data-augmentation, but

also by including all the source data for the training with the pseudo-labeled target data.

**Interactions with $\mathscr{N}$:** $\mathscr{C}$ interacts multiplicatively with $\mathscr{N}$. This means that reducing $\mathscr{C}$ limits the negative impact of a high $\mathscr{N}$ by multiplication. More concretely, this indicates that Model Regularization can help to reduce the negative impact of noise on the final UDA re-ID performance. Particularly, using the source data, for Pseudo-Labeling, would allow to reduce the impact of the noise in the pseudo-labels, by reducing overfitting of the training data.

### 4.4.3 Domain Discrepancy term: $\mathscr{D}\mathscr{D}$

The Domain Discrepancy term $\mathscr{D}\mathscr{D}$ of our new learning bound for Pseudo-Labeling UDA re-ID (Eq. 4.38 in Sec. 4.3.6) can be decomposed into the the $\mathscr{H}\Delta\mathscr{H}$-distance between the source and target domains $d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}},\mathscr{D}^{\mathscr{T}}\right)$, the ideal joint error $\lambda$, and the quantity $1-\alpha$ representing the weight put on the source samples in the empirical risk $\hat{\epsilon}_\alpha$ (Eq. 4.15). As we recall, the best $\alpha$ cannot be easily estimated in a UDA problem. It is also impossible to act on the ideal joint error $\lambda$ that is set given the class of hypothesis $\mathscr{H}$ and the source and target domains. Therefore, to reduce the expected target risk $\epsilon_{\mathscr{T}}(\hat{h})$, $d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}},\mathscr{D}^{\mathscr{T}}\right)$ should remain as low as possible. For this, in practice, the Domain Discrepancy between the source and target domains is penalized when learning the UDA re-ID model i.e. Domain Alignment is performed.

**Interactions with $\mathscr{N}$:** $\mathscr{D}\mathscr{D}$ interacts with $\mathscr{N}$, indirectly through a trade-off controlled by the $\alpha$ term. More specifically, the more the training relies on the source-guidance ($\alpha$ "close to" 0, which reduces the negative impact of $\rho_{-1}$ and $\rho_{+1}$ in $\mathscr{N}$), the more $\mathscr{D}\mathscr{D}$ increases. Therefore, the more the training relies on source-guidance, the more important it is to reduce the Domain Discrepancy by Domain Alignment in order not to degrade the UDA re-ID performance.

### 4.4.4 Loss Bound: M

The loss bound M of our new learning bound for Pseudo-Labeling UDA re-ID (Eq. 4.38) is highlighted by the use of the "0-M" loss. It can increase multiplicatively the negative impact of $\mathscr{N}$ and $d_{\mathscr{H}\Delta\mathscr{H}}\left(\mathscr{D}^{\mathscr{S}},\mathscr{D}^{\mathscr{T}}\right)$ in $\mathscr{D}\mathscr{D}$. M should therefore be controlled and limited, for example by favoring the use of Bounded Loss to learn re-ID.

### 4.4.5 The deduced good practices

Our bound analysis allows us to understand more clearly the Pseudo-Labeling training paradigm using the source samples. Furthermore, it also allows to deduce general good practices to improve cross-domain performance of UDA re-ID, detailed hereafter:

- **Source-guided Learning with Domain Alignment**: it consists in reducing overfitting by using the labeled source data for re-ID feature learning, particularly of the pseudo-label errors. It should be performed jointly with Domain Alignment, which constrains the feature encoder to align the source and target domains in the feature space, in order to alievate the Domain Discrepancy negative impact on the performance.
- **Bounded Loss**: it consists in reducing the amplification of the Domain Discrepancy and Complexity terms by M, by using a Bounded Loss for re-ID feature learning.
- **Outlier Filtering**: it consists in reducing the impact of the noise term by filtering outliers in pseudo-labeled target samples.
- **Model Regularization**: it consists in reducing noise overfitting by regularizing the model.

According to our theoretical analysis, following these good practices should improve performance on the target domain and make the best use of source data when training with pseudo-labels. In Fig. 4.1, we illustrate the classical Pseudo-Labeling cyclic training, as well as, a Pseudo-Labeling training where all good practices are followed: Source-guided Pseudo-Label training with good practices. Good practices are general and represented in green on the figure. They can be implemented in various ways that we discuss in the next section.

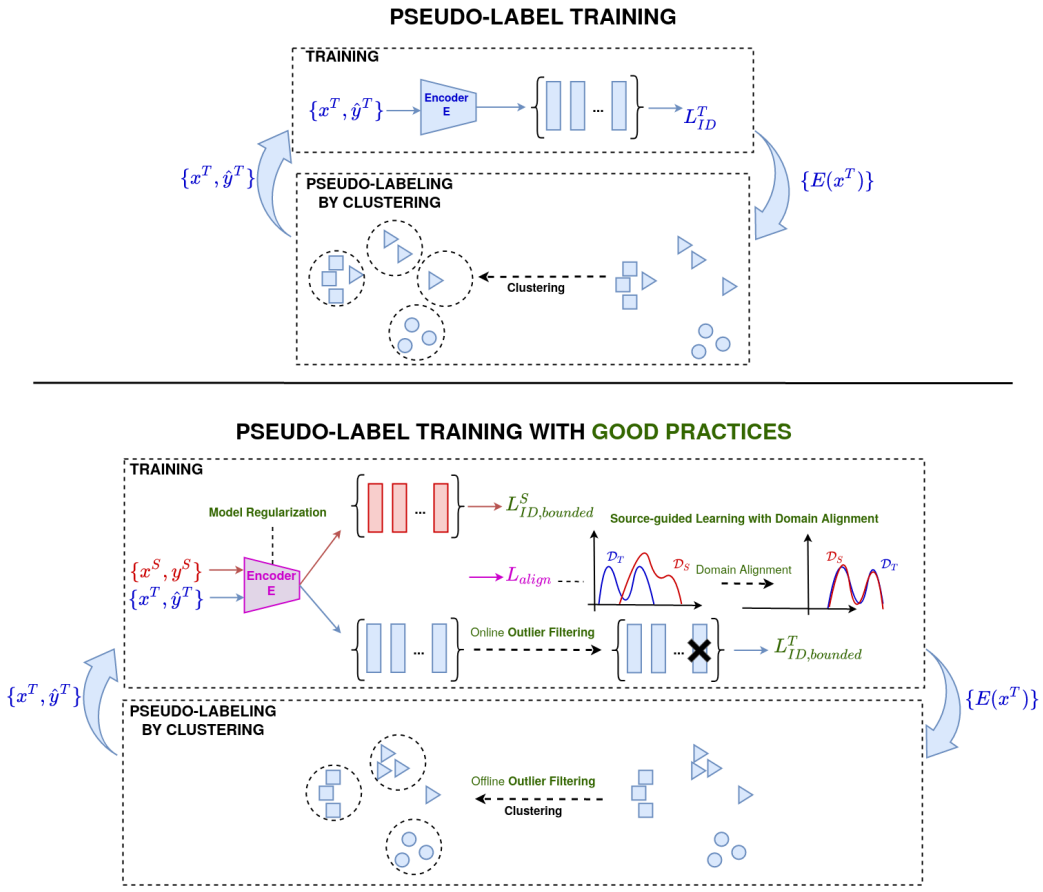


Figure 4.1: Classical Pseudo-Label Training is illustrated at the top of the figure. At the bottom, the Pseudo-Label Training with good practices is illustrated. good practices are derived from analysis of our new learning bound for Pseudo-Labeling, to improve UDA re-ID performance. These good practices, when followed and implemented in Pseudo-Labeling method, aim at improving the re-ID performance on the target domain. These good practices, represented in green on the figure, are: *Source-guided Learning with Domain Alignment*, using a *Bounded Loss* for re-ID learning, *Model Regularization* and *Outlier Filtering* (performed offline and/or online). $\mathrm{L}^{\mathcal{S}}_{\mathrm{ID},bounded}$ and $\mathrm{L}^{\mathcal{T}}_{\mathrm{ID},bounded}$ are bounded loss functions resp. defined for the source $(\{x^{\mathcal{S}}, y^{\mathcal{S}}\})$ and target $(\{x^{\mathcal{T}}, \hat{y}^{\mathcal{T}}\})$ samples to learn re-ID features. $\mathrm{L}_{align}$ is a loss that penalizes a domain discrepancy measure between the source and the target distributions (resp. $\mathscr{D}^{\mathcal{S}}$ and $\mathscr{D}^{\mathcal{T}}$) in the similarity feature space

## 4.5 Implementing good practices

This section aims at validating by experiments good practices derived from the theoretical learning bound in Sec. 4.4. For this, we propose different ways to implement good practices in a Pseudo-Labeling method of interest in Sec. 4.5.1. Then, we introduce a set of state-of-the-art baselines in Sec. 4.5.2, that follow different good practices, to which we will implement the missing ones.

### 4.5.1 Implementing good practices into a Pseudo-Labeling framework

Table 4.1: Summary of the relationships between the theoretical analysis conducted throughout the chapter and the implementation of good practices in Pseudo-Labeling UDA re-ID frameworks. Examples refer to existing state-of-the-art solutions to enforce good practices in the framework.

| Theory | Good practices | Examples of implementations |
|---|---|---|
| Noise Reduction ( $\searrow \mathscr{N}$ ) | Outlier Filtering (Pseudo-Label Refinery) | DBSCAN, Asymetric Co-Teaching , Online Outlier Filtering, etc |
| Overfitting Reduction ( $\searrow \mathscr{C}$ ) | Model Regularization<br>Source-guided Learning | Mean Teaching, Feature Memory Bank, etc |
| Domain Discrepancy Reduction ( $\searrow \mathscr{DD}$ ) | Domain Alignment | Maximum Mean Discrepancy,<br>Adversarial Domain Adaptation, etc |
| Bound Limitation ( $M < +\infty$ ) | Bounded re-ID Loss | Loss Thresholding,<br>Triplet Loss with Normalized Features, etc |

In order to experimentally validate good practices derived from the theory for UDA re-ID Pseudo-Labeling, they have to be implemented into Pseudo-Labeling methods of interest. Hereafter, different possible ways to implement them in Pseudo-Labeling UDA re-ID methods are proposed. The links between the theory, good practices and their implementations are summarized in Tab. 4.1.

**Source-guided Learning with Domain Alignment** Source-guided Learning consists in learning the re-ID model with the source samples, in addition to the target samples. Simultaneously with learning with the source data, good practices include Domain Alignment. More specifically, to establish our bound, we modeled the re-ID problem as a verification task on pairs of images. Therefore, the binary classifier takes as input a vector measuring the similarity between the pair of image features. We call this space the *similarity space*. Domain Alignment should be performed in this space according to the theory.

In the framework of UDA, hyperparameter $\alpha$ must be set in the empirical loss function $\hat{\epsilon_\alpha}$ (Eq. 4.15). Without a priori knowledge, we arbitrarily put the same weight on the source and target samples contribution in the Source-guided Pseudo-Labeling re-ID loss function, i.e. $\alpha = 0.5$ for all the experiments. Indeed, setting $\alpha$ to 0.5 is generally done in UDA re-ID frameworks using the source data ([42, 30]).

For Domain Alignment in the similarity space, minimization of the Maximum Mean Discrepancy (MMD) ([47] has already been used for UDA re-ID in other works ([96]). Note that Domain Alignment can be implemented differently, like for instance with a 2-layer Domain Adversarial Neural Network (DANN ([39])). The Optimal Transport can also be considered to perform Domain Alignment based on the Sinkhorn algorithm ([22]) (with a regularization parameter set to 1).

Following some existing works ([96]) for UDA re-ID, MMD is chosen with the same Gaussian kernel settings. The MMD loss is therefore directly added to the re-ID loss function.

**Outlier Filtering** Outlier Filtering aims at improving the number of correctly pseudo-labeled samples, by pseudo-label refinery or by discarding erroneous pseudo-labeled samples from the training stages. It can be done in an *offline* way, by updating the pseudo-labels by clustering and by discarding them from the whole training set during this clustering stage i.e. before the training stages. It can also be done in an online way, at the batch level, i.e. during the training stages.

To perform offline filtering, DBSCAN [32] is generally used to update the pseudo-labels and discard the erroneous ones. This clustering algorithm considers as outliers the samples belonging to clusters with a number of samples inferior to a number specified as a hyperparameter.

As for *online* filtering, state-of-the art approaches introduced advanced techniques such as Asymmetric Co-Teaching (ACT) ([143]) or online label propagation ([143, 175]). Since performance improvements have been shown when using DBSCAN ([175]), this suggests that online outlier filtering should be useful in addition to performing offline outlier filtering with DBSCAN. Therefore, Outlier Filtering should include offline and online outlier filtering. However, most existing online filtering strategies are method-specific and can easily increase the compu-

tation cost and resources needed for a Pseudo-Labeling approach they are added to: for example a teacher network ([143]) or feature and label memory banks ([175]).

Therefore, we propose a more simple and general online outlier filtering strategy based on an outlier detection statistical test: the Tukey Criterion ([11]). This criterion is applied on the loss values. Indeed, the intuition behind using the loss values for filtering, as for ACT ([143]), is that uncommonly high values of the loss function are more likely to correspond to outliers. Then ACT filters out the pseudo-labeled samples having loss values beyond a preset threshold, using a teacher network. Therefore, computing the Tukey Criterion gives us a loss threshold above which the target domain samples are considered as outlier and thus are discarded from the batch.

It is also possible to propose a lighter version of ACT, where the teacher is the model itself. Basically, we filter out the target samples corresponding to the top-$p$% ($p \in \mathbb{N}$) highest loss values in every batch.

In the rest of the chapter, experiments are conducted with DBSCAN and the Tukey Criterion for the implementation of Outlier Filtering. The confidence coefficient of rejecting the null hypothesis of Tukey Criterion is set to 0.05.

**Bounded Loss** The Bounded Loss practice aims at controling the loss bound M. An easy way to control the loss bound would be to use a bounded loss for re-ID learning. However, in practice, the "0-M" loss cannot be used for re-ID learning with gradient descent, because it is not differentiable. Indeed, other classical losses are used for re-ID such as the Cross-Entropy classification loss, the Triplet Loss or the Contrastive loss ([42]).

In order to avoid changing the loss functions of the UDA re-ID method, which are sometimes at the core of the UDA re-ID approach, we propose a more flexible strategy to control their bound, that we call *Thresholding*. When using Triplet Loss, that optimizes directly on the distances between features, Thresholding practice consists in normalizing the features, which consequently bounds the distances on the unit norm ball, and thus the loss function. Moreover, for the other loss functions, Thresholding practice thresholds the values of the loss above a defined threshold. The samples associated to the loss values above the threshold are discarded from the batch and therefore will not be back-propagated for parameter

updates by gradient descent. In order not to introduce a new arbitrarily-set hyper-parameter for this loss threshold, the threshold obtained by the Tukey Criterion used for Outlier Filtering, described in the previous paragraph, can be reused to choose the loss threshold.

While UDA re-ID classically uses unbounded losses, for the classification task, bounded classification losses robust to label noise have been designed. In particular, Ghosh et al. ([44]) show that the *Mean Absolute Error* (MAE) is an unbounded loss which improves empirically the classification performance when the labels are corrupted by different amount of noises. Moreover, it can be easily computed as the L1 penalization of the difference between 1 and the predicted probability for the ground-truth class by the model. Therefore, this could be another candidate to implement Bounded Loss.

In the good practice experiments, the Thresholding strategy is used as a Bounded Loss.

**Model Regularization** Model Regularization aims at limiting the model complexity to prevent overfitting. In practice, Model Regularization can be performed with general regularization techniques, such as Weight Decay [66] that penalizes the L2 norm of the model parameters. Or they can specifically be chosen and designed to be robust to noisy pseudo-labels, such as Mutual Mean Teaching ([41]).

Model Regularization is to our knowledge followed by all Pseudo-Labeling UDA re-ID methods which always use weight decay. Moreover, specific Model Regularization implementations, designed to be robust to noisy labels are generally inherent to the framework design as in MMT ([41]): changing them or performing an ablation of them would completely distort the method. For these reasons, in future experiments, we choose to keep as they are the Model Regularization of the Pseudo-Labeling baselines of interest.

### 4.5.2 State-of-the-art baselines

We focus on four different state-of-the-art Pseudo-Labeling UDA re-ID baselines. As it will be detailed hereafter, our choice has been motivated by the fact that these baselines follow a varied inventory of good practices. Moreover, they correspond to recent UDA re-ID methods, with state-of-the art performances and an available code for reproducibility.

**UDAP ([120])**. UDAP ([120]) is one of the first approaches using pseudo-labels for the UDA re-ID. In the learning process, the source data is only used to pretrain the feature encoder which initializes the first pseudo-labels. This approach uses DBSCAN as a clustering algorithm and therefore follows Outlier Filtering, more particularly offline Outlier Filtering, outside of the training loops. Moreover, UDAP ([120]) minimizes the Triplet Loss, which is unbounded by definition. Finally, UDAP ([120]) does not use any specific regularization on the model parameters apart from the classical weight decay.

**MMT ([41])**. Like UDAP ([120]), MMT ([41]) does not exploit the source data when training with the pseudo-labels. All the target samples are pseudo-labeled by k-means clustering algorithm and used for training: MMT ([41]) does not perform Outlier Filtering at all. MMT ([41]) optimizes both a Triplet Loss and a Cross-Entropy loss, with hard and soft pseudo-labels: these loss functions are also unbounded. Finally, MMT ([41]) relies on the mutual learning paradigm and a mean teacher updated by an exponential moving average of the student model parameters. In semi-supervised learning, Mean Teachers ([125]) as well as Mutual Learning ([162]) are seen as consistency regularization by ensembling. Therefore, we can consider the Mutual Mean Teaching of MMT ([41]) as a specific Model Regularization technique, used in addition to weight decay.

**SpCL ([42])**. Unlike the UDAP ([120]) and MMT ([41]) approaches, SpCL ([42]) leverages the source data during the training phases, in addition to the pseudo-labeled target data. However, SpCL ([42]) does not perform Domain Alignment to reduce the domain discrepancy in the feature space. Moreover, a contrastive loss is optimized, which is an unbounded loss function. In addition to the Outlier Filtering performed by DBSCAN, SpCL ([42]) further filters outliers with an additional cluster reliability criterion: Reliable Clusters. It consists in performing 2 clustering of the target dataset by DBSCAN, with 2 different density hyperparameter, and discard samples inconsistent between the 2 clusterings. Finally, SpCL ([42]) uses a moving average of class-centroid and instance feature memory bank to compute the class centroids. As MMT ([41]), it can be viewed as consistency regularization by temporal ensembling of feature, and therefore as a Model Regularization module.

Table 4.2: Dataset composition

| Dataset | # train IDs | # train images | # test IDs | # gallery images | # query images |
|---|---|---|---|---|---|
| Market ([172]) | 751 | 12,936 | 750 | 16,364 | 3,368 |
| Duke ([111]) | 702 | 16,522 | 702 | 16,364 | 2,228 |
| PersonX ([123]) | 410 | 9,840 | 856 | 17,661 | 30,816 |
| MSMT ([140]) | 1,041 | 32,621 | 3,060 | 82,161 | 11,659 |
| Vehicle-ID ([82]) | 13,164 | 113,346 | 800 | 7,332 | 6,532 |
| Veri ([86]) | 575 | 37,746 | 200 | 49,325 | 1,678 |
| VehicleX ([86]) | 1,362 | 192,150 | N.A. | N.A. | N.A. |

In summary, the existing Pseudo-Labeling approaches for UDA re-ID do not follow all good practices (as summarized in Tab. 4.3)

### 4.5.3 Implementation details.

We reused the codes[1] [2] [3] given by the baseline authors as well as the same implementation details (learning rate, architecture,...) in their respective paper ([120, 41, 42, 31]). For MMT, different architectures and hyperparameter values for the number of clusters for k-means are used in the paper. To allow fair comparison with other frameworks, we choose the ResNet-50 ([51]) architecture and report the best performance of MMT among the number of clusters tested in the paper. For cross-dataset benchmarks that are not available in their paper, we use the number of ground-truth clusters of the target training set for the number of clusters. We use 4 x 24Go NVIDIA TITAN RTX GPU for all of our experiments.

### 4.5.4 Cross-dataset benchmarks.

We study the effectiveness of good practices on the re-ID performance by computing and reporting the mean Average Precision (mAP in %) and rank-1 (%) on the target test sets for different cross-dataset adaptation tasks. It is important to have a variety of source and target datasets, since as the theory suggests, the bound depends on properties specific to the datasets (proportion of source/target samples in the dataset, domain discrepancy...). *Person re-ID* is evaluated on the large re-ID

[1] https://github.com/LcDog/DomainAdaptiveReID
[2] https://github.com/yxgeee/MMT
[3] https://github.com/yxgeee/SpCL

dataset MSMT17 ([140]) (*MSMT*): used as the target domain, it offers a challenging adaptation task due to its large number of images and identities in its gallery (cf. dataset statistics in Tab. 4.2). We also use Market-1501 ([172]) (*Market*) as the target domain, using the synthetic dataset PersonX as the source domain. *PersonX* ([123]) is composed of synthetic images generated on Unity with different types of person appearances, camera views and occlusions. Then we also report classical benchmarks between Market and DukeMTMC-reID ([111]) (*Duke*). *Vehicle re-ID* is less reported than Person re-ID for UDA re-ID benchmarking. However, we find it interesting to test our module on a different kind of object of interest and on a potentially different domain discrepancy. We use for this task the *Vehicle-ID* ([82]), Veri-776 ([86]) (*Veri*) datasets and the synthetic vehicle dataset *VehicleX* ([97]).

Table 4.3: This table represents good practices already followed in the four original state-of-the-art frameworks of interest: UDAP ([120]), MMT ([41]), SpCL ([42]), compared to their respective version following all good practices (w/ all good practices). The missing good practices implementations are represented in bold and green. WD = Weight Decay ; MMD = Maximum Mean Discrepancy

| Method | Bounded loss | Outlier Filtering | Source-guided Learning | Domain Alignment | Model Regularization |
|---|---|---|---|---|---|
| UDAP ([120]) | × | DBSCAN (✓) | × | × | WD (✓) |
| UDAP ([120]) w/ all good practices | **Thresholding (✓)** | DBSCAN (✓) **Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | WD (✓) |
| MMT ([41]) | × | × | × | × | WD (✓) Mutual Mean Teaching (✓) |
| MMT ([41]) w/ all good practices | **Thresholding (✓)** | **DBSCAN (✓)** **Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | WD (✓) Mutual Mean Teaching (✓) |
| SpCL ([42]) | × | DBSCAN (✓) Reliable Clusters (✓) | ✓ | × | WD (✓) Hybrid Memory (✓) |
| SpCL ([42]) w/ all good practices | **Thresholding (✓)** | DBSCAN (✓) Reliable Clusters (✓) **Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | WD (✓) Hybrid Memory (✓) |

To experimentally validate these good practices derived from the theory, as well as their generalization to different UDA re-ID frameworks, we propose to complement the presented baselines with missing good practices implemented as in Sec. 4.5.1.

Table 4.4: Comparison of original baselines with their corresponding version in which the missing good practices have been implemented (w/ all good practices) for different person cross-dataset benchmarks. mAP and rank-1 are reported in %

| Method | Market→MSMT | | PersonX→Market | | PersonX→MSMT | | Market→Duke | | Duke→Market | |
|---|---|---|---|---|---|---|---|---|---|---|
| | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 |
| UDAP ([120]) | 12.0 | 30.6 | 48.4 | 68.4 | 10.5 | 26.3 | 50.1 | 70.2 | 55.3 | 78.1 |
| UDAP ([120]) w/ all good practices | **20.9** | **47.0** | **67.1** | **70.8** | **14.9** | **36.1** | **63.1** | **77.3** | **69.4** | **86.5** |
| MMT ([41]) | 22.9 | 49.2 | 70.8 | 66.8 | 16.9 | 38.5 | 65.1 | 78.0 | 71.2 | 87.7 |
| MMT ([41]) w/ all good practices | **25.1** | **52.9** | **73.4** | **88.0** | **18.9** | **43.2** | **67.9** | **82.0** | **75.5** | **88.8** |
| SpCL ([42]) | 25.7 | 53.4 | 72.2 | 86.1 | 22.1 | 47.7 | 68.3 | 82.5 | 76.1 | 89.8 |
| SpCL ([42]) w/ all good practices | **27.0** | **53.9** | **74.1** | **88.6** | **23.0** | **47.8** | **70.6** | **83.8** | **78.0** | **91.4** |

Table 4.5: Comparison of original baselines and their corresponding version in which the missing good practices have been implemented (w/ all good practices) for different vehicle cross-dataset benchmarks. mAP and rank-1 are reported in %

| Method | VehicleID→Veri | | VehicleX→Veri | |
|---|---|---|---|---|
| | mAP | rank-1 | mAP | rank-1 |
| UDAP ([120]) | 35.6 | 74.1 | 35.0 | 75.9 |
| UDAP ([120]) w/ all good practices | **37.1** | **75.2** | **37.2** | **77.1** |
| MMT ([41]) | 36.4 | 74.2 | 36.3 | 75.8 |
| MMT ([41]) w/ all good practices | **37.9** | **80.1** | **37.7** | **81.2** |
| SpCL ([42]) | 37.6 | 79.7 | 37.4 | 81.0 |
| SpCL ([42]) w/ all good practices | **39.0** | **83.4** | **39.2** | **83.8** |

## 4.6 Experimental results

In Sec. 4.6.1, we will show, by extensive experiments on various person and vehicle re-ID cross-datasets benchmarks, that implementing the missing good practices in these baselines can in fact improve their UDA re-ID performance. By ablative study in Sec.4.6.2, we analyze the contribution to the UDA re-ID performance of each followed good practices. Finally, in Sec. 4.6.3, we show by additional experiments, consistency of UDA re-ID performance improvement when good practices are implemented differently but still present.

### 4.6.1 Improving UDA re-ID baselines by following good practices

In Tab. 4.4 and Tab. 4.5, we reported the mAP in % obtained after adding the missing good practices in different baselines (see Tab. 4.3) resp. for person cross-dataset and vehicle cross-dataset benchmarks. We notice that for all frameworks UDAP ([120]), MMT ([41]) and SpCL ([42]), following good practices consistently improves performance on all person and vehicle adaptation tasks. More specifically, we notice that this improvement depends on how many elements of good practices are added in relation to the original baseline. Apparently, the more missing good practices a method has, the more it is likely to benefit from following all good practices. For example, UDAP ([120]), which follows only 2 good practices, gains +8.9 p.p. mAP on Market→MSMT, while SpCL ([42]), with the greatest number of good practices already followed, gains +2.1 p.p. mAP by following the missing good practices. Consistently, on Vehicle-ID→Veri, UDAP ([120]) gains +1.5 p.p. mAP, while SpCL ([42]) gains +0.9 p.p. mAP by enforcing the missing good practices. Overall, enforcing good practices systematically improves performance whatever the number of missing good practices a framework follows. The next section aims at analyzing more thoroughly these performance gains with an ablative study.

### 4.6.2 Ablation study on good practices

As seen in Sec. 4.6.1, following all good practices in a Pseudo-Labeling UDA re-ID framework improves cross-dataset performance. To quantify the contribution of each good practices individually, as well as to understand their interactions, we carried out on 2 cross-dataset benchmarks (PersonX→Market and Market→MSMT) an ablative study for each framework: UDAP ([120]), MMT ([41]) and SpCL

([42]) resp. in Tab. 4.6, Tab. 4.7 and Tab. 4.8. Variant 1, 2, 3, 4 resp. designate the addition of Bounded Loss, Outlier Filtering and Domain Alignment to the baselines.

Table 4.6: Ablative study comparing UDAP ([120]) to a version of UDAP ([120]) where each practices have been followed individually. The missing and newly-added good practices are represented in bold and green. mAP are reported in % for two cross-dataset UDA re-ID tasks: PersonX→Market and Market→MSMT.

| Variants | Bounded Loss | Outlier Filtering | Source-guided Learning | Domain Alignment | PersonX→Market | Market→MSMT |
|---|---|---|---|---|---|---|
| UDAP ([120]) | × | DBSCAN (✓) | × | × | 48.4 | 12.0 |
| 1 | **Thresholding (✓)** | DBSCAN (✓) | × | × | 50.3 | 14.3 |
| 2 | × | DBSCAN (✓)<br>**Tukey Online Filtering (✓)** | × | × | 59.7 | 16.9 |
| 3 | × | DBSCAN (✓) | ✓ | × | 53.1 | 14.9 |
| 4 | × | DBSCAN (✓) | ✓ | **MMD (✓)** | 60.0 | 16.8 |
| UDAP ([120])<br>w/ all good practices | **Thresholding (✓)** | DBSCAN (✓)<br>**Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | **67.1** | **20.9** |

Table 4.7: Ablative study comparing MMT ([41]) to a version of MMT ([41]) where each practices have been followed individually. The missing and newly-added good practices are represented in bold and green. mAP are reported in % for two cross-dataset UDA re-ID tasks: PersonX→Market and Market→MSMT

| Variants | Bounded Loss | Outlier Filtering | Source-guided Learning | Domain Alignment | PersonX→Market | Market→MSMT |
|---|---|---|---|---|---|---|
| MMT ([41]) | × | × | × | × | 70.8 | 22.9 |
| 1 | **Thresholding (✓)** | × | × | × | 71.7 | 24.5 |
| 2 | × | **DBSCAN (✓)**<br>**Tukey Online Filtering (✓)** | × | × | 72.2 | 24.3 |
| 3 | × | × | ✓ | × | 72.9 | 24.1 |
| 4 | × | × | ✓ | **MMD (✓)** | 73.0 | 24.7 |
| MMT ([41])<br>w/ all good practices | **Thresholding (✓)** | **DBSCAN (✓)**<br>**Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | **73.4** | **25.1** |

**Bounded Loss** We first compare the original frameworks with their variant 1 (in Tab. 4.6, Tab. 4.7 and Tab. 4.8), which bounds the re-ID loss function by Thresholding, we note that the Bounded Loss good practices allows an improvement of the re-ID performance for all frameworks on the two cross-dataset UDA re-ID tasks. For PersonX→Market, variant 1, with the Bounded Loss good practices followed, improves the original framework resp. by +1.9 p.p., +0.9 p.p., +0.7 p.p. and +0.6 p.p. mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]). Consistently, for Market→MSMT, variant 1, improves the original framework resp. by +2.3 p.p., +1.6 p.p., +1.1 p.p. and +0.9 p.p. mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]). This is in line with our analysis of the learning bound, from which we deduce the Bounded Loss as a good practices improving the UDA re-ID performance. Moreover, we also notice that a framework with fewer Outlier Filtering and Model

Table 4.8: Ablative study comparing SpCL ([42]) to a version of SpCL ([42]) where each practices have been followed individually. The missing and newly-added good practices are represented in bold and green. mAP are reported in % for two cross-dataset UDA re-ID task: PersonX→Market and Market→MSMT.

| Variants | Bounded Loss | Outlier Filtering | Source-guided Learning | Domain Alignment | PersonX→Market | Market→MSMT |
|---|---|---|---|---|---|---|
| SpCL ([42]) | × | DBSCAN (✓)<br>Reliable Clusters (✓) | × | × | 72.2 | 25.7 |
| 1 | **Thresholding (✓)** | DBSCAN (✓)<br>Reliable Clusters (✓) | ✓ | × | 72.9 | 26.8 |
| 2 | × | DBSCAN (✓)<br>Reliable Clusters (✓)<br>**Tukey Online Filtering (✓)** | ✓ | × | 73.3 | 26.6 |
| 4 | × | DBSCAN (✓)<br>Reliable Clusters (✓) | ✓ | **MMD (✓)** | 73.2 | 26.5 |
| SpCL ([42])<br>w/ all good practices | **Thresholding (✓)** | DBSCAN (✓)<br>Reliable Clusters (✓)<br>**Tukey Online Filtering (✓)** | ✓ | **MMD (✓)** | **74.1** | **27.0** |

Regularization good practices, seems to benefit from a better improvement of its performance when Bounded Loss good practices are added: the mAP increase is higher for UDAP ([120]) compared to MMT ([41]), higher for MMT ([41]) compared to SpCL ([42]) and higher for SpCL ([42]) than for SpCL ([42]). We reckon it is in line with our theoretical analysis: it should be caused by the multiplicative interaction in the learning bound between the loss bound M and the complexity term that is further reduced with better Model Regularization for MMT ([41]) and SpCL ([42]), as well as the multiplicative interaction between M and the Domain Discrepancy term $\mathscr{DD}$ which is further reduced by Domain Alignment for SpCL ([42]).

**Outlier Filtering** Variant 2 (in Tab. 4.6, Tab. 4.7 and Tab. 4.8), adding Outlier Filtering, improves performance over the original frameworks. For PersonX→Market, Outlier Filtering improves the original framework by resp. +11.3 p.p., +1.1 p.p., +1.4 p.p and +0.7 p.p mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]). Consistently, for Market→MSMT, variant 2, improves the original framework resp. by +4.9 p.p., +1.4 p.p., +0.9 p.p. and +0.8 p.p. mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]). Therefore, it seems to confirm that Outlier Filtering, and more generally reducing the noise probabilities, is a good practices for Pseudo-Labeling UDA re-ID. What's more, we also notice that this enhancement is more significant for UDAP ([120]) and less for frameworks with better Model Regularization (MMT ([41]), SpCL ([42]), SpCL ([42])). Again, we guess that this result accounts for the multiplicative interaction in the learning bound between the Noise term $\mathscr{N}$ and the Complexity term $\mathscr{C}$ reduced by Model Regularization.

**Source-guided Learning without Domain Alignment without Domain Alignement** By using the source samples to learn re-ID (variant 3 of UDAP ([120]) and MMT ([41]) resp. in Tab. 4.6 and Tab. 4.7), performance get improved over the original frameworks. For PersonX→Market, Source-guided Learning improves the original framework resp. by +4.7 p.p and +2.1 p.p. mAP for UDAP ([120]) and MMT ([41]). Consistently, for Market→MSMT, Source-guided Learning improves the original framework resp. by +2.9 p.p. and +1.2 p.p. mAP for UDAP ([120]) and MMT ([41]). This performance improvement is consistent with our learning bound analysis, which states that using the source samples should help improve the UDA re-ID performance.

**Source-guided Learning with Domain Alignment with Domain Alignement** Adding Domain Alignment (variant 4 for UDAP ([120]), MMT ([41]) and SpCL ([42]) in Tab. 4.6, Tab. 4.7, and Tab 4.8) further improves performance. For PersonX→Market, Source-guided Learning with Domain Alignment improves the original framework resp. by +11.6 p.p., +2.2 p.p. and +1.0 p.p mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]). Consistently, for Market→MSMT, adding Domain Alignment increases the performance resp. by +4.8 p.p., +1.8 p.p. and +0.8 p.p. mAP for UDAP ([120]), MMT ([41]) and SpCL ([42]) . It is therefore more effective to reduce the domain gap with Domain Alignment when using the source samples to learn re-ID.

**Model Regularization** Model Regularization is one of the good practices derived from our theoretical analysis. As discussed previously in Sec. 4.3.6, we did not perform an ablation of Model Regularization from the Pseudo-Labeling approaches since their regularization techniques generally define the core of these approaches. Yet, to experimentally confirm the role of Model Regularization as a good practices, it is possible to compare UDAP ([120]) w/ all good practices to MMT ([41]) w/ all good practices. Indeed, MMT ([41]) w/ all good practices corresponds to UDAP ([120]) w/ all good practices to which we add the Mutual Mean Teaching model regularization. Therefore, MMT ([41]) w/ all good practices improves the UDA re-ID performance resp. by +6.3 p.p. and +4.2 p.p. mAP for PersonX→Market and Market→MSMT. Model Regularization, specifically designed for preventing noise overfitting (like MMT ([41])), seems therefore to be a key good practices to signifi-

cantly boost the UDA re-ID performance.

### 4.6.3 Experiments with other implementations of good practices

This section adds additional experiments to show that the derived good practices are still consistent when implemented differently, particularly for Outlier Filtering and Domain Alignment for which other strategies exist.

Table 4.9: Impact on the UDA re-ID performance of MMT ([41]) when using different Outlier Filtering strategies. mAP are reported in % for the cross-dataset task PersonX→Market. The percentage of outliers in a batch after filtering, averaged over all the training iterations, are also reported in %

| Outlier Filtering | PersonX→Market | Average of outliers per batch |
|---|---|---|
| × | 70.8 | 27.2 |
| DBSCAN | 71.1 | 19.3 |
| Tukey | 71.8 | 10.1 |
| DBSCAN + Tukey | 72.2 | 7.2 |
| top-5% | 71.1 | 18.4 |
| top-10% | 71.5 | 10.2 |
| top-20% | 70.9 | 3.9 |
| DBSCAN + top-10% | 72.0 | 8.0 |

**Changing the Outlier Filtering strategy** Here we change the way Outlier Filtering is performed in the framework. Experiments are conducted for MMT ([41]) which does not perform any Outlier Filtering. Different Outlier Filtering strategies are evaluated: DBSCAN, Tukey, DBSCAN + Tukey, and top-5/10/20% introduced in Sec. 4.5.1. The re-ID cross-dataset performance (mAP) is reported in Tab. 4.9, as well as the average percentage of outliers in a batch during all the training. This quantity is computed by counting the pairs of data with same pseudo-label yet different ground-truth labels as well as those with different pseudo-labels yet same ground-truth labels. First, we notice that every Outlier Filtering strategy improves the original framework. The performance improvement ranges from +0.1 p.p. mAP with top-20% to +1.4 p.p. mAP with DBSCAN + Tukey. These experiments seem to confirm that there are different ways of enforcing the Outlier Filtering good practices in a Pseudo-Labeling UDA re-ID framework.
Moreover, by analyzing the average percentage of outliers per batch, we can conclude that every Outlier Filtering strategy effectively reduces the number of

outliers, as expected. Even if, in general, a lower percentage of outliers may be correlated to a better final re-ID performance, the top-20% strategy seems to be an exception. Indeed, the top-20% strategy filters out more outliers than other strategies but does not offer the best re-ID performance in the end. We suppose that Outlier Filtering strategies may discard valuable training samples, particularly hard positive/negative samples, which are more likely to be selected as outlier whereas they represent valuable information to learn ID discriminative representation.

Table 4.10: Impact on the UDA re-ID performance when using different Domain Alignment strategies. mAP are reported in % for the cross-dataset task PersonX→Market on UDAP ([120])

| Domain Alignment | PersonX→Market |
|---|---|
| × | 48.4 |
| MMD | 60.0 |
| Domain Adversarial Neural Network ([39]) | 60.7 |
| Optimal Transport (Sinkhorn ([22]) ) | 61.1 |

**Changing the Domain Alignment criterion** We also conducted more experiments with UDAP ([120]) for PersonX→Market where the Domain Alignment criterion is changed when performing Source-guided Learning with Domain Alignment, with some implementations introduced in Sec. 4.5.1. In Tab. 4.10, for PersonX→Market, whatever the Domain Alignment strategy, we notice performance improvements of UDAP ([120]) ranging from +11.6 p.p. mAP for MMD to +12.7 p.p. mAP for Optimal Transport. Again, experiments show flexibility on the way used to enforce the Domain Alignment good practices.

**Changing the Bounded Loss** In the previous experiments, to follow the Bounded Loss good practices, we chose to bound the re-ID implemented in the framework using the threshold computed with the Tukey Criterion applied on the loss values.
In the following experiments, the Triplet Loss of UDAP ([120]) is replaced by the MAE loss introduced in Sec. 4.5.1. Experiments are conducted for PersonX→Market. In Tab. 4.11, using the MAE Loss to implement the Bounded Loss good practices, the cross-domain re-ID performance is improved by +2.1 p.p. mAP. As it has been shown for the Thresholding, following the Bounded Loss good practices, with the

MAE loss, indeed improves the cross-domain re-ID performance. This indicates that the Bounded Loss is a general good practices that can be implemented with different strategies.

Table 4.11: Impact on the UDA re-ID performance when using different Bounded Loss strategies. mAP are reported in % for the cross-dataset task PersonX→Market on UDAP ([120]).

| Bounded Loss | PersonX→Market |
|---|---|
| × | 48.4 |
| Thresholding | 50.3 |
| Mean Absolute Error (MAE) ([44]) | 50.5 |

## 4.7 Conclusion and discussion

In this chapter, we derived general good practices for pseudo-labeling UDA through a new theoretical framework which encompasses the relationship between the source knowledge and the target pseudo-label errors. The proposed theoretical view, throughout a learning bound on the cross-domain performance, provides insight on how Source-Guided Pseudo-Labeling works and highlights the conditions ensuring the performance improvement of good practices as well as the links that bind them to explain this improvement. Our work could have a broader impact by providing insight not only for UDA re-ID, but also for other UDA tasks for which pseudo-labeling methods prevail. For now, the work in this thesis has focused on improving cross-domain performance by exploiting useful information from the source. The work in the state of the art is also focused on improving cross-domain performance. However, it is important to remember that the cross-domain performance drop problem is parts of the practical challenge of deploying a re-ID system. However, the proposed work does not take into account this aspect of the problem: could we deploy pseudo-labeling methods and obtain as good performances as obtained in an academic context ? It is on this essential aspect of the cross-domain problem that the rest of this thesis work focuses.

# Chapter 5

# Automatic Source-Guided Selection of Pseudo-Labeling Hyperparameters

## 5.1 Introduction

The objective of this chapter is to address the cross-domain problem from a practical and complementary aspect of performance: the deployability of UDA re-ID pseudo-labeling methods. More concretely, we ask ourselves if the performances obtained by methods designed in the framework of academic research could be transferred to the real world with such good performances. This chapter highlights a double problem that pseudo-labeling approaches face, and that limits the deployability of their performances:

- The cross-domain performances of these approaches are sensitive to the choice of clustering HyperParameter (HP): the ideal HP value changes according to the considered target domain.

- The constraint of the absence of annotation on the target data in the UDA counter makes it impossible to use classical techniques of HP selection by using a set of annotated validations coming from the same domain: the current methods assume the reuse of the same HP value whatever the target domain.

Once the problem has been set in Sec. 5.2, a new approach to automatically select clustering hyperparameter (HP) values, adapted to the target domain, is proposed in the form of two contributions:

- Theoretical modeling and insights that shed light on the conditions under which source-based validation is relevant for the UDA re-ID clustering task are provided (Sec. 5.3).

- A novel method to automate the selection of clustering HP used by pseudo-labeling approaches is proposed: HyperParameters Automated by Source & Similarities (HyPASS). It consists in (i) a Source-Guided automatic HP tuning performed before each clustering phase and (ii) a conditional domain alignment of feature similarities with source ID-discriminative features applied during the training phase to improve HP selection (Sec. 4.4).

Extensive experiments on commonly used and challenging re-ID tasks for people and vehicles, as well as ablative studies, show that HyPASS can be integrated into the best state-of-the-art pseudo-labeling methods and improves consistently its cross-domain re-ID performance compared to one with a less well-chosen HP value using empirical setting (Sec. 5.5 and Sec. 5.6).

This chapter has been published in IEEE Access journal [31].
Industrial applications are also considered, with a patent application under evaluation. In the context of an industrial collaboration on cross-domain cow re-ID, a paper has been submitted and accepted at CVPRW 2022, and a journal paper has been submitted to IJCV (under review).

## 5.2 Motivation

### 5.2.1 Pseudo-Labeling UDA re-ID: the cross-domain performance sensibility to clustering HP

Recently, pseudo-labeling approaches have proven to be the best UDA methods to learn ID-discriminative features for the target domain [181] [120] [42]. For this purpose, these methods rely on generating artificial labels for the target unlabeled training data. Due to the open-set nature of the re-ID UDA task, pseudo-labels are generally generated by clustering the target training samples [120, 41, 42]. To this end, it is necessary to specify values for the HP that set the clustering algorithm. Density-based clustering algorithms [7, 95, 32] are the most widespread in the UDA re-ID literature. In particular, DBSCAN [32] is used for its effectiveness in a

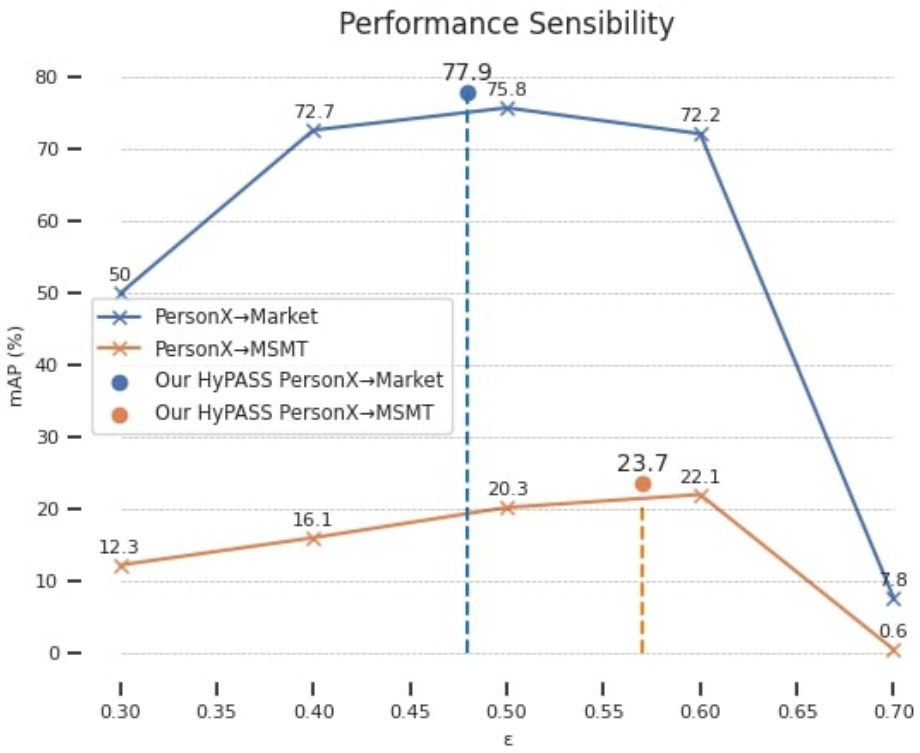


Figure 5.1: Performance sensibility of the best state-of-the-art methods SpCL [42] with respect to parameter ϵ (the maximum neighborhood distance) of DBSCAN [32] for two different cross-dataset experiments. HyPASS consistently ensures a better HP choice. HyPASS performs cyclic pseudo-labeling HP tuning and for more clarity we only represent in the Figure the final performance for ϵ value found for the last training stage.

large majority of pseudo-labeling approaches, including the best performing ones [42, 157]. For DBSCAN, one HP to set is ϵ, defined as the maximum neighborhood distance. Despite the development of approaches robust to noise in pseudo-labels [41, 42], their final performance is still quite sensitive to the choice of ϵ. In Fig. 5.1, there is a limited range of ϵ values for which performance of SpCL [42], the best state-of-the-art methods, remain near 'optimal' and not very sensitive. Indeed, given a cross-dataset task, for example PersonX→Market, these values seem condensed in a range around $\epsilon = 0.5$, where performance reaches a mAP of 75.8%. However, if ϵ is set to 0.6, performance drops to 72.2%. For $\epsilon = 0.7$, the performance drop is even sharper: down to 7.8%.

Therefore, selecting a suitable value for this critical HP is crucial to obtain the

best performance. This behavior is not specific to DBSCAN and the same can be said for HP k of k-means (this will be discussed later in Sec. 5.6.3 with Fig. 5.4).

### 5.2.2 Choosing the right clustering HP for Pseudo-Labeling: a challenge for UDA re-ID

The lack of labels for the target data makes this selection non-trivial in the UDA context. Unlike the supervised setting, it is impossible to form a labeled validation set to do HP tuning with a re-ID performance metric on the target domain (mAP, rank-1...). The state-of-the-art for UDA re-ID [120, 42] sets these critical pseudo-labeling HP (like $\epsilon$) by validation on one adaptation task (e.g. PersonX→MSMT) with a *labeled* target validation data set, then uses this empirical value for other adaptation tasks. This empirical setting strategy assumes that a value selected for HP from one adaptation task transfers well to another one. However, this assumption only holds to a certain extent and, to our knowledge, there is no rule to know in advance how well this value transfers to a new task in the UDA setting. In Fig. 5.1, by using this strategy for SpCL [42] method, with the best value $\epsilon$ on PersonX→MSMT ($\epsilon = 0.6$), we get a mAP of 72.2% on the PersonX→Market task. However, if we had chosen $\epsilon = 0.5$ we could have obtained a better mAP of 75.8%. This indicates that empirical setting has its limits and that a task-specific choice of HP would be more desirable in order to get maximum performance of the pseudo-labeling method. Again, these remarks also apply to other clustering algorithms (see [41] and Fig. 5.4 for k-means). Moreover, the clusters depend on the learned feature representation. As the feature representation varies through learning, this HP choice might even be better if we could cyclically adjust its value to the learned feature representation before each pseudo-labeling updates by clustering.

### 5.2.3 The lack of robust clustering HP choice strategy for Pseudo-Labeling UDA re-ID

Pseudo-labeling methods generally exploit a source-trained model to initialize pseudo-identity labels for target data. The pseudo-labels are generated by clustering the target data feature representations obtained by this model. Some works on pseudo-labeling define their own strategy to assign labels to target data based, for example, on similarity to a selected set of prototypes [153, 181, 90, 130, 79,

155]. Most pseudo-labeling methods are built on a self-learning iterative paradigm which alternates between (i) optimization for target re-ID feature learning with the lastly optimized model on target images and (ii) pseudo-label prediction (pseudo-labeling) by feature clustering [120, 159, 62, 124, 156, 183, 143, 14, 41, 157, 163, 183, 103, 160, 144]. Most of these works improve the classical self-learning algorithm on not overfitting the pseudo-label errors, by using teacher-student or ensemble of expert models [41, 163, 157] while other approaches focus on designing efficient sample selection and outlier detection strategies [143, 14]. More robust frameworks are also designed by optimizing losses based on distance distributions [62, 87], by leveraging local features [38], intra-inter camera features [142, 80], the labeled source samples [30], multiple cluster views [35] or attention-based model [61], or by mixing pseudo-labels with domain-translation methods [156, 124, 183, 16], online pseudo-label refinery strategy, temporal ensembling and label propagation [160, 167] or meta learning [144]. A recent approach, SpCL [42], proposed self-contrastive learning during the training phase, by leveraging the source and target samples. Most of the above-mentioned pseudo-labeling methods, including the best and most recent ones, use DBSCAN to pseudo-label the target training samples [120, 159, 62, 124, 156, 183, 96, 143, 14, 163, 183, 41, 157]. They are all possibly affected by the clustering sensibility to HP, as it is shown in [120] and illustrated in Fig. 5.1, where performance of the best state-of-the-art methods, SpCL, depends on the choice of a critical HP. Other approaches, using less common clustering algorithms, also seem concerned (shown later in Sec. 5.6.3 with Fig. 5.4 for k-means). Moreover, to our knowledge, they all choose a fixed empirical value to set this HP, which remains the same no matter the adaptation task, and through all the pseudo-labeling cycles. Performance of these approaches may suffer from this restricted HP setting. Our contribution aims at overcoming those limiting aspects by integrating a new automatic and cyclic HP selection phase into the pseudo-labeling cycle. Our contribution aims to be general so that it can be easily integrated and improve any existing or future pseudo-labeling approach.

### 5.2.4 Existing solutions for Hyperparameter Selection for UDA classification

As HP selection in the UDA setting has been studied, to our knowledge, only for the classification task, we focus on the related work for this task. In UDA

classification, HP selection remains a major problem. Many approaches in UDA classification use the same strategy as UDA re-ID pseudo-labeling methods: the empirical setting of HP values, used on different cross-dataset adaptation tasks [126, 107, 113, 101]. Manually labeling a part of the target dataset to make a validation set [56] is out of the UDA context. The use of a source validation set [39, 106] offers biased estimation of the classification target expected risk because of the domain discrepancy. Importance weighting methods [121, 89, 20] tackle this issue by weighting the estimated risk with source samples but they still suffer from high variance estimation. The recent work [150] improves these approaches and proposes an importance-weighted cross-validation in the feature space to reduce the source estimator variance. However, two major aspects prevent its application for HP selection of the pseudo-labeling UDA clustering. First, it requires the estimation of probability densities of the source and target distributions (in the feature space). Cyclically integrated in a pseudo-labeling framework, these densities should be re-estimated before each update of the pseudo-labels by clustering. This would be harder to integrate in any pseudo-labeling methods, computationally expensive and the ratio of estimated densities could increase approximation errors. Finally, the approach is adapted for classification problems only, which differs from the clustering task.
To our knowledge, there is no general work on clustering HP selection adapted to UDA pseudo-labeling. That's why we recast the theory behind these source leveraging approaches [121, 89, 20, 150] to fit the clustering task. Moreover, in order to better integrate it into pseudo-labeling approaches, our approach takes a new turn compared to those ones, by avoiding estimation of importance weights: we propose to optimize the model for domain alignment in the feature similarity space with source ID-discriminative features to improve the estimation with a source validation set by reducing its variance.

## 5.3 Theoretical Grounds of Hyperparameter Selection for Clustering in UDA re-ID

The selection of HP $\lambda \in \mathbb{R}^m, m \in \mathbb{N}^*$ consists in finding the value $\lambda^* \in \mathbb{R}^{n_\lambda}$ that minimizes a defined expected risk. Unlike the models learnable parameters, HP are not directly learnt during the training loop of a machine learning pipeline. A typ-

ical strategy to estimate $\lambda^*$ is model selection: among a set of candidate models defined by different HP values, we choose the one that gives the lowest empirical risk. This strategy is not applicable with the UDA setting because target annotations are not available. Moreover, as discussed in Sec. 5.2.4, existing approaches (for classification) are not directly adapted for re-ID. The goal of this section is thus to give theoretical leads that will give us more insights about two questions: How do the source data bias the target risk estimation? How to overcome this bias? We first introduce notations and the problem formulation (Sec. 5.3.1). Then we define the expected risk to optimize for the clustering task (Sec. 5.3.2), in order to deduce an empirical estimate based on the source data (Sec. 5.3.3). Finally, a focus is given to the variance of this estimate to better understand how to improve HP selection by reducing it (Sec. 5.3.4). For this, we first show that the variance can be reduced by reducing the domain discrepancy between the source and target in the feature similarity space (Sec. 5.3.5). Then we give theoretical analysis on the pairwise ratio, showing that with reasonable assumptions, the source empirical risk can be used directly to do efficient HP selection (Sec. 5.3.5).

### 5.3.1 Problem Formulation and Notations

#### Offline vs Online Cyclic HP tuning for clustering

If we focus on the iterative pseudo-labeling paradigm, we can note that the learned feature representation changes during each training phase of an iterative cycle. Since the pseudo-labels are updated by clustering in this representation space, we intuitively expect the optimal clustering hyperparameter value to change when this representation changes (as it will be shown empirically in Sec. 5.6.2). The model selection is classically done via an evaluation criterion on the downstream task (in our case the re-ID as a retrieval task). Proceeding in this way necessarily implies training completely with selected HP values, evaluating (with re-ID metrics such as mAP) and repeating again and again. This would thus make the selection computationally expansive (a training time analysis is given in Sec. 5.6.3). To overcome this, our idea is to perform an online model selection directly at the clustering task level, at each iterative cycle.

**Modeling the clustering task**

As introduced in Sec. 5.3, the first step is to define the expected risk to be minimized w.r.t $\lambda$ for HP selection. This expected risk $\mathscr{R}_{\mathscr{L},p}$ (defined in [127]) is defined in relation to the unknown distribution of data characterized by the probability density $p$ and a cost function $\mathscr{L}$ which depends on our underlying task: a clustering task for our problem. A clustering is considered "good" when it generates pseudo-labels related to the ground-truth identity labels. Our idea is therefore to model this clustering task as a verification problem. For this, let's suppose that the re-ID data are i.i.d and come from an unknown joint distribution given by the density $p(x,x',r)$ defined on $\chi \times \chi \times \{-1,1\}$ where $\chi \subseteq \mathbb{R}^{n_\chi}, n_\chi \in \mathbb{N}$ represents the set of images for which $r=1$ if $x$ and $x'$ have the same ID and $r=-1$ otherwise. Thus, the goal is to find a clustering function $\mathrm{C}_\lambda$ which is expected to classify all the $m \in \mathbb{N}$ pairs of images in a set $\mathrm{X} = ((x_i, x'_i)_{1\leq i\leq m})$ as their respective ground truth labels are $\mathrm{R} = (r_i)_{1\leq i\leq m}$. We also assume that clusters are predicted from a measure of similarities between elements in the set. For the set X, the pairwise similarities are given by $\mathrm{S}(\mathrm{X}) = (s(x_i, x'_i))_{1\leq i\leq m}$, where $s : \chi \times \chi \to \mathbb{R}$ is a given similarity function. Therefore, $\mathrm{C}_\lambda$ is a $\mathbb{R}^m \to \{-1,1\}^m$ function.

### 5.3.2 Similarity-Based Clustering Risk Minimization

By definition, following previous notations, the expected risk $\mathscr{R}_{\mathscr{L},p}$ for the clustering task can be seen as a function of $\lambda$:

$$\mathscr{R}_{\mathscr{L},p}(\lambda) \triangleq \int_{\mathrm{X},\mathrm{R}} \mathscr{L}\big(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X})), \mathrm{R}\big) p(\mathrm{X},\mathrm{R}) d\mathrm{X} d\mathrm{R}\,, \tag{5.1}$$

where $p(\mathrm{X},\mathrm{R})$ is a joint probability density defined on $(\chi \times \chi)^m \times \{-1,1\}^m$.
The UDA setting for the clustering task does not involve only one distribution associated to its density $p$, but two distributions related to the source $\mathscr{S}$ and the target $\mathscr{T}$. Their joint probability densities are noted respectively $p^{\mathscr{S}}(\mathrm{X},\mathrm{R})$ and $p^{\mathscr{T}}(\mathrm{X},\mathrm{R})$. To perform source-based HP selection, we need to link the target expected risk $\mathscr{R}_{\mathscr{L},p^{\mathscr{T}}}$ defined by Eq. 5.1 with $p^{\mathscr{S}}$.

### 5.3.3 Similarity Importance-Weighted Risk

We consider the re-ID UDA context with the target and source distributions defined above. Our goal is to link the target expected risk (Eq. 5.1) with $p^{\mathscr{S}}$. By devel-

oping the target expected risk, we have:

$$\begin{aligned}\mathscr{R}_{\mathscr{L},p^{\mathscr{T}}}(\lambda) &= \int_{\mathrm{X,R}} \mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X})),\mathrm{R})\, p^{\mathscr{T}}(\mathrm{X},\mathrm{R})\, d\mathrm{X} d\mathrm{R} \\ &= \int_{\mathrm{X,R}} \frac{p^{\mathscr{T}}(\mathrm{X},\mathrm{R})}{p^{\mathscr{S}}(\mathrm{X},\mathrm{R})} \mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X})),\mathrm{R})\, p^{\mathscr{S}}(\mathrm{X},\mathrm{R})\, d\mathrm{X} d\mathrm{R} \\ &= \int_{\mathrm{X,R}} w(\mathrm{X},\mathrm{R}) \mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X})),\mathrm{R})\, p^{\mathscr{S}}(\mathrm{X},\mathrm{R})\, d\mathrm{X} d\mathrm{R}.\end{aligned} \tag{5.2}$$

The pairwise weight ratio $w$ is defined as:

$$w(\mathrm{X},\mathrm{R}) \triangleq \frac{p^{\mathscr{T}}(\mathrm{X},\mathrm{R})}{p^{\mathscr{S}}(\mathrm{X},\mathrm{R})}. \tag{5.3}$$

Then we can define the pairwise weighted risk as:

$$\mathscr{R}_{\mathscr{L},w}(\lambda) \triangleq \int_{\mathrm{X,R}} w(\mathrm{X},\mathrm{R}) \mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X})),\mathrm{R})\, p^{\mathscr{S}}(\mathrm{X},\mathrm{R})\, d\mathrm{X} d\mathrm{R}. \tag{5.4}$$

From Eq. 5.4, we can deduce the associated pairwise weighted empirical risk which is an unbiased estimator of $\mathscr{R}_{\mathscr{L},p^{\mathscr{T}}}(\lambda)$ with finite source samples:

$$\hat{\mathscr{R}}_{\mathscr{L},w}(\lambda) = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} w(\mathrm{X}_i,\mathrm{R}_i) \mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}(\mathrm{X}_i)),\mathrm{R}_i), \tag{5.5}$$

where $\{(\mathrm{X}_i,\mathrm{R}_i)\}_{1\leq i\leq \mathrm{N}}, \mathrm{N}\in\mathbb{N}^*$ are samples from $p^{\mathscr{S}}(\mathrm{X},\mathrm{R})$.

### 5.3.4 Variance of the estimator

Even if the estimator given by Eq. 5.5 is unbiased, a high variance can add noise to HP selection with source samples. Before giving an expression of the estimator's variance, we define the exponential in base 2 of the Rényi divergence (called Rényi divergence in the rest of the chapter for simplicity) of order $\alpha \geq 0$, $\alpha \neq 1$ between the source and target distribution described by the densities $p^{\mathscr{S}}$ and $p^{\mathscr{T}}$ as:

$$
\begin{aligned}
d_{\alpha}(p^{\mathcal{T}}||p^{\mathcal{S}}) &\triangleq \left(\int_{\mathrm{X,R}} \frac{p^{\mathcal{S}}(\mathrm{X},\mathrm{R})^{\alpha}}{p^{\mathcal{T}}(\mathrm{X},\mathrm{R})^{\alpha-1}} d\mathrm{X} d\mathrm{R}\right)^{\frac{1}{\alpha-1}} \\
&= \left(\int_{\mathrm{X,R}} w(\mathrm{X},\mathrm{R})^{-\alpha} p^{\mathcal{T}}(\mathrm{X},\mathrm{R}) d\mathrm{X} d\mathrm{R}\right)^{\frac{1}{\alpha-1}} \\
&= \left(\mathop{\mathbb{E}}_{(\mathrm{X,R})\sim p^{\mathcal{T}}}[w(\mathrm{X},\mathrm{R})^{-\alpha}]\right)^{\frac{1}{\alpha-1}}.
\end{aligned}
\tag{5.6}
$$

Let Y be $\mathrm{Y} = w(\mathrm{X},\mathrm{R})\mathscr{L}(\mathrm{C}_{\lambda}(\mathrm{S}(\mathrm{X})),\mathrm{R})$ for $(\mathrm{X},\mathrm{R}) \sim p^{\mathcal{S}}(\mathrm{X},\mathrm{R})$. Using the lemma 2 from Cortes *et al.* [20] and with the definition of $\hat{\mathscr{R}}_{\mathscr{L},w}$ (Eq. 5.5), we can get a bound on the variance of $\hat{\mathscr{R}}_{\mathscr{L},w}$:

$$
\begin{aligned}
\mathrm{V}ar(\mathrm{Y}) &= \mathop{\mathbb{E}}_{(\mathrm{X,R})\sim p^{\mathcal{S}}}[\mathrm{Y}^2] - \mathop{\mathbb{E}}_{(\mathrm{X,R})\sim p^{\mathcal{S}}}[\mathrm{Y}]^2 \\
&\leq d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})\mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}(\lambda)^{1-\frac{1}{\alpha}} - \mathop{\mathbb{E}}_{(\mathrm{X,R})\sim p^{\mathcal{S}}}[\mathrm{Y}]^2 \\
&\leq d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})\mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}(\lambda)^{1-\frac{1}{\alpha}} - \mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}(\lambda)^2 \\
\mathrm{V}ar(\hat{\mathscr{R}}_{\mathscr{L},w}) &\leq \frac{d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})\mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}(\lambda)^{1-\frac{1}{\alpha}} - \mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}(\lambda)^2}{\mathrm{N}},
\end{aligned}
\tag{5.7}
$$

This bound on the empirical risk variance confirms the intuition that the more source (validation) samples we have, the lesser is the variance. However, in practice, the amount of labeled source samples is limited. Therefore we cannot act on this constant in order to improve our estimation. However, this bound on the empirical risk variance also shows that the greater the $d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})$, the greater the variance of the estimator. In order to control this variance, and therefore improve the use of the pairwise weighted empirical risk estimator for model selection, it is necessary to control $d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})$ which measures the domain discrepancy between $p^{\mathcal{T}}$ and $p^{\mathcal{S}}$ according to the Rényi divergence. Moreover, reducing this divergence should make the estimation less sensible to the number of source validation samples according to Eq. 5.7.

### 5.3.5 Addressing the variance and weight ratio

#### Using feature similarity

The input space (images) is high-dimensional. Therefore, $d_{\alpha+1}(p^{\mathcal{T}}||p^{\mathcal{S}})$ is more likely to be greater (and thus the variance of the estimator given by Eq. 5.7) than the divergence between probability distributions in a lower-dimensional feature space (

as stated in Sec. 4.2 of [150]). Indeed, the pairwise weight ratio can more likely grow to infinity since $p^{\mathscr{S}}$ when $p^{\mathscr{T}} \neq 0$ is more likely to be 0. Moreover, a feature space induced by a learnable feature encoder could allow us to reduce the divergence by penalizing it during the learning phase.
Usually in re-ID, a feature space is learned so that a given similarity function used in this space can measure ID relatedness between images. Therefore, we introduce a learnable feature encoder $f_\theta : \chi \to \mathbb{R}^{n_f}, n_f \in \mathbb{N}$ parametrized by $\theta \in \mathbb{R}^p, p \in \mathbb{N}$ and redefine $s : \mathbb{R}^{n_f} \times \mathbb{R}^{n_f} \to \mathbb{R}$. We also define $\mathrm{S}_f$ the feature similarity function with respect to $f_\theta$ such as $\mathrm{S}_f(\mathrm{X}) = (s(f_\theta(x_i), f_\theta(x'_i)))_{1 \leq i \leq m}$. Thus, $\mathrm{S}_f$, projects the set of images X into a new set $\mathrm{S} \in \mathbb{R}^m$, in a space we call the feature similarity space. Let $p^{\mathscr{S}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R})$ (resp. $p^{\mathscr{T}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R})$) be the feature similarity distribution densities of $\mathscr{S}$ (resp. $\mathscr{T}$) induced by $\mathrm{S}_f$ and defined on $\mathbb{R}^m \times \{-1,1\}^m$. We consider this space as our new input space for computing the risks and therefore if we note

$$w_{\mathrm{S}_f}(\mathrm{S},\mathrm{R}) = \frac{p^{\mathscr{T}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R})}{p^{\mathscr{S}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R})}, \tag{5.8}$$

with analogous definitions and notations, we deduce the pairwise similarity weighted empirical risk $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$:

$$\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}(\lambda) = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} w_{\mathrm{S}_f}(\mathrm{S}_i,\mathrm{R}_i)\mathscr{L}(\mathrm{C}_\lambda(\mathrm{S}_i),\mathrm{R}_i), \tag{5.9}$$

where $\{(\mathrm{S}_i,\mathrm{R}_i)\}_{1 \leq i \leq \mathrm{N}}, \mathrm{N} \in \mathbb{N}$ are samples from $p^{\mathscr{S}}_{\mathrm{S}_f}$.
In practice, we have directly access to sets of pairwise image samples $\{(\mathrm{X}_i,\mathrm{R}_i)\}_{1 \leq i \leq \mathrm{N}}$ defined above and we use $\mathrm{S}_f$ to get $\{(\mathrm{S}_i,\mathrm{R}_i)\} = \{(\mathrm{S}_f(\mathrm{X}_i),\mathrm{R}_i)\}$.
According to Eq. 5.4, $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$ is an unbiased estimator of the expected target risk $\mathscr{R}_{\mathscr{L},p^{\mathscr{T}}_{\mathrm{S}_f}}$, that we can use to do HP selection of $\lambda$ with source labeled samples. We expect this new estimator to be better for HP selection. Indeed, we expect it to have a lower variance than due to the lower domain discrepancy in this learnable low-dimensional feature space (as stated in Sec. 4.2 of [150]):

$$\mathrm{V}ar(\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}) \leq \mathrm{V}ar(\hat{\mathscr{R}}_{\mathscr{L},w}). \tag{5.10}$$

In addition, the pairwise data samples being i.i.d. (see Sec. 5.3.1), the pairwise similarities are i.i.d. too and therefore the densities in the feature similarity space can be written as:

$$\begin{cases} p^{\mathcal{S}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R}) = p^{\mathcal{S}}(\mathrm{R})\Pi_{i=1}^{m} p^{\mathcal{S}}_{\mathrm{S}_f}(\mathrm{S}_i|\mathrm{R}_i) \\ p^{\mathcal{T}}_{\mathrm{S}_f}(\mathrm{S},\mathrm{R}) = p^{\mathcal{T}}(\mathrm{R})\Pi_{i=1}^{m} p^{\mathcal{T}}_{\mathrm{S}_f}(\mathrm{S}_i|\mathrm{R}_i)\,. \end{cases} \tag{5.11}$$

In Eq. 5.3.5, since $p^{\mathcal{S}}(\mathrm{R})$ and $p^{\mathcal{T}}(\mathrm{R})$ are fixed by the domain distributions and are independent from E, we assume that E can be learned to penalize the conditional domain discrepancy (i.e. the the divergence between the conditional distributions) in the feature similarity space in order to improve HP selection with our estimator $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$.

**Computing the pairwise weight ratio**

To sum up, our goal is to do HP selection of $\lambda$ by minimizing $\mathscr{R}_{\mathscr{L},p^{\mathcal{T}}}$ (Eq. 5.4) w.r.t $\lambda$. For this, we established the expression of the pairwise weighted empirical risk estimator $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$ with source samples (Eq. 5.9). This estimator will be improved by learning $f$ to penalize the conditional domain discrepancy in the feature similarity space. Using $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$ requires to compute $w_{\mathrm{S}_f}$. As mentioned in Sec. 5.2.4, unlike importance weighted risk estimation approaches for UDA classification, we do not wish to estimate the pairwise weight ratio in a pseudo-labeling framework: this would require estimating the probability density of this ratio at each new pseudo-labeling step. This would be computationally expensive. Moreover, the quotient of estimated probabilities in the ratio could increase approximation errors and therefore add noise to the risk estimate. To avoid computing pairwise weight ratio, it would be desirable that we can do HP selection using the source empirical risk $\hat{\mathscr{R}}_{\mathscr{L},p^{\mathcal{S}}_{\mathrm{S}_f}}$.

To do relevant HP selection using $\hat{\mathscr{R}}_{\mathscr{L},p^{\mathcal{S}}_{\mathrm{S}_f}}$ instead of $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$, it is therefore necessary that $\operatorname{argmin}_\lambda \hat{\mathscr{R}}_{\mathscr{L},p^{\mathcal{S}}_{\mathrm{S}_f}}(\lambda) \approx \operatorname{argmin}_\lambda \hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}(\lambda)$. In other words, this ensures that selecting the best $\lambda$ with $\hat{\mathscr{R}}_{\mathscr{L},p^{\mathcal{S}}_{\mathrm{S}_f}}$ is the same as selecting the best $\lambda$ with $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}$.
Given the expression of $\hat{\mathscr{R}}_{\mathscr{L},w_{\mathrm{S}_f}}(\lambda)$ (Eq. 5.9), a direct sufficient condition to ensure this can be:

$$\forall 1 \leq i \leq \mathrm{N}, w_{\mathrm{S}_f}(\mathrm{S}_i,\mathrm{R}_i) = c, c \in \mathbb{R}+. \tag{5.12}$$

In practice, Eq. 5.12 can be satisfied by using the whole source validation set as a unique pair $(\mathrm{S},\mathrm{R})$ to do HP selection. This will be part of our framework design

choice as discussed later in Sec. 5.4.1 in what we call One-clustering evaluation.

To summarize, these theoretical considerations show us that to select HP λ from the source examples, it is sufficient to minimize the source empirical risk, and that we minimize the conditional domain discrepancy in the feature similarity space w.r.t E.

## 5.4 Source-Guided Selection of Pseudo-Labeling Hyperparameters and Similarity Alignment

We wish to apply the theory discussed above and integrate it into a pseudo-labeling algorithm. For this purpose, we propose a novel method integrated into the classical iterative pseudo-labeling paradigm [120]: HyperParameters Automated by Source & Similarities (HyPASS). Fig. 5.2 gives an overview of the incremented method. HyPASS consists in integrating a new clustering HP selection phase (Auto HP TUNING) from a source validation set before each clustering update and optimizing the model to minimize the conditional feature similarity domain discrepancy $L^{cond}_{align}$. In this part, we give more details about this two major novelties.

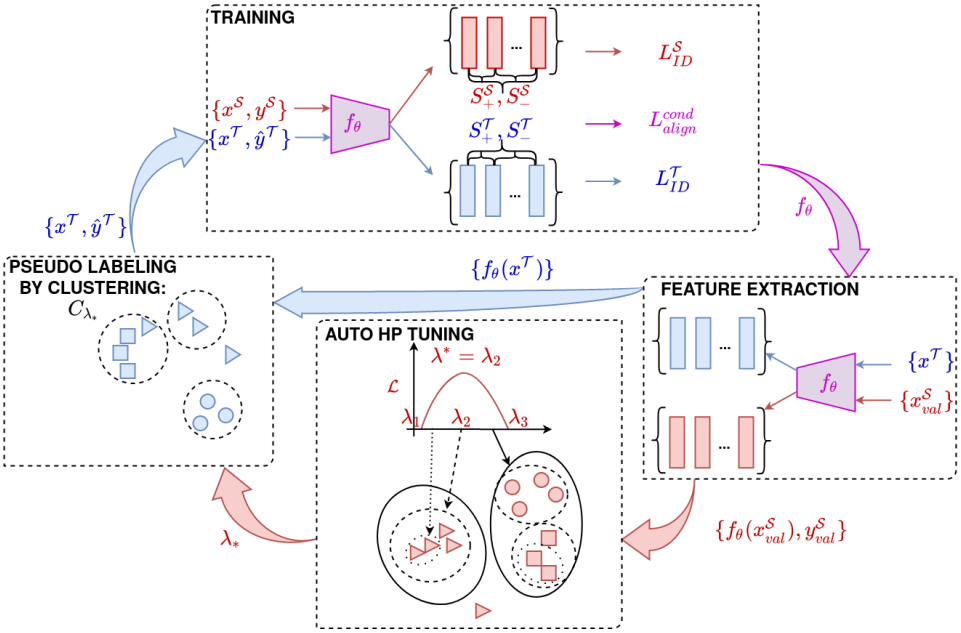


Figure 5.2: Our HyperParameter Automated by Source & Similarities (HyPASS) cyclically integrated in iterations of a classical pseudo-labeling framework.

### 5.4.1 Automatic Clustering HP Tuning

Our method proposes a new step of automatic selection of clustering HP $\lambda$. This selection is cyclic because it takes place at each cycle before the update of the pseudo-labels, in order to adapt the selected HP to the representation learned by E.

**One-clustering evaluation.** We suppose we have access to a separate labeled source validation set $\mathrm{D}^{\mathcal{S}}_{val}$ of $\mathrm{N}^{\mathcal{S}}_{val}$ samples. We also assume that HP search is restricted to a finite size set $\Lambda \subset \mathbb{R}^{n_\lambda}$. Given a clustering criterion $\mathscr{L}$ and a HP value $\lambda$ to evaluate, HP tuning phase uses the source empirical risk with samples from $\mathrm{D}^{\mathcal{S}}_{val}$. Remember that to satisfy Eq. 5.12, we should use the whole set of validation samples and on a one-clustering evaluation of the associated risk. Moreover, it can be very computationally expensive to do multiple clustering steps to evaluate a unique HP value, and $\mathrm{N}^{\mathcal{S}}_{val}$ can be 'too small' to split $\mathrm{D}^{\mathcal{S}}_{val}$ into different subsets for clustering. Therefore, we decide to only perform one clustering on the full set $\mathrm{D}^{\mathcal{S}}_{val}$ to evaluate one parameter value of $\lambda$ with the source empirical risk. At the end of this step, we keep the value $\lambda$ that gives the lowest empirical risk value.

### 5.4.2 Learning with conditional domain alignment of feature similarities.

#### Learning features for re-ID

From the pseudo-labels, the model is trained to minimize a loss function $\mathrm{L}^{\mathcal{T}}_{\mathrm{ID}}$ in order to learn an ID-discriminative feature representation on the target domain. This loss function can be for example the cross entropy loss, the triplet loss, a contrastive loss function or the sum of several of these terms. Besides, we also wish this representation to be ID-discriminative on the source domain by optimizing a loss function $\mathrm{L}^{\mathcal{S}}_{\mathrm{ID}}$ with the labeled source samples. Intuitively, we motivate this choice in order not to degrade the discriminativeness of the representation on the target domain, while optimizing the feature similarity alignment between source and target.

#### Domain Discrepancy

Reducing the domain discrepancy in the conditional similarity feature space is a key aspect to reduce the variance when using the source empirical estimation (as

shown in Sec. 5.3.5). Given a differentiable domain alignment criterion $L_{align}$ (e.g., Maximum Mean Discrepancy (MMD) [113]), we optimize the domain alignment in the conditional feature similarity space given by the formula:

$$L_{align}^{cond} = L_{align}(S_{+}^{\mathcal{S}}, S_{+}^{\mathcal{T}}) + L_{align}(S_{-}^{\mathcal{S}}, S_{-}^{\mathcal{T}}), \tag{5.13}$$

where $S_{+}^{\mathcal{S}}$, $S_{+}^{\mathcal{T}}$, $S_{-}^{\mathcal{S}}$ and $S_{-}^{\mathcal{T}}$ are the similarity score (e.g.: the cosine similarity) computed between features of, resp., positive pairs of the source, positive pairs of the target, negative pairs of the source and negative pairs of the target. Minimizing this term aligns intra-cluster similarity distributions but also inter-cluster similarity distributions between domains.

**Global criterion**

The total loss $L_{total}$ is given by:

$$L_{total} = L_{\text{ID}}^{\mathcal{T}} + L_{\text{ID}}^{\mathcal{S}} + L_{align}^{cond}. \tag{5.14}$$

Note that we choose not to weight the different loss terms in $L_{total}$ in order not to introduce new additional HP in the UDA context. Indeed, experiments in Sec. 5.5 will show that this loss choice already allows to get performance improvements from HyPASS in various UDA benchmarks.

### 5.4.3 General pseudo-code of HyPASS

In addition to Fig. 5.2, we propose in the Algo. 3 a pseudo-code for training a pseudo-labeling re-ID UDA framework by using HyPASS. The proposed automatic HP tuning from source data (AUTO HP-TUNING) called by Algo. 3 is detailed in Algo. 4 introduced by our approach.
Algo. 3 describes the whole HyPASS training paradigm. A model is first initialized (INITIALIZATION) to predict the first pseudo-labels for the target training set. Then the algorithm iterates cyclically through a FEATURE EXTRACTION phase with the actual model for the source validation set and the target training set. Then during the AUTO HP-TUNING phase a value for $\lambda^*$ is automatically selected by maximizing a clustering quality criteria. Then this HP value is used to pseudo-label/cluster the target training features during the PSEUDO-LABELING phase. Then the model is fine-tuned with the source training set and the pseudo-labeled target training using

HyPASS loss function (see Eq. 5.14). Algo. 4 further details the AUTO HP-TUNING phase, where the algorithm iterates through different HP values proposed by a HP selection strategy or function which are used to pseudo-label the source validation set and compute with the source label a clustering quality metric to be maximized.

**Algorithm 3** HyperParameters Automated by Source & Similarities (HyPASS)

**Input:** Labeled source training set $\mathrm{D}^{\mathcal{S}}$
**Input:** Labeled source validation set $\mathrm{D}^{\mathcal{S}}_{val}$: $\mathrm{D}^{\mathcal{S}}_{val} \cap \mathrm{D}^{\mathcal{S}} = \varnothing$
**Input:** Unlabeled target data $\mathrm{D}^{\mathcal{T}}$
**Input:** Clustering/Pseudo-labeling function $\mathrm{C}_\lambda$ with HP $\lambda$
**Input:** HP list $\Lambda$
**Input:** Clustering/Pseudo-Labeling quality metric $\mathscr{L}$ (to maximize)
**Input:** Loss Functions for Training: $\mathrm{L}^{\mathcal{S}}_{\mathrm{ID}}$, $\mathrm{L}^{\mathcal{T}}_{\mathrm{ID}}$, $\mathrm{L}_{align}$
**Input:** Number of training epochs $\mathrm{N}_{epoch}$
**Input:** Feature encoder E
**INITIALIZATION:**
Compute $\mathrm{S}^{\mathcal{S}}$, $\mathrm{S}^{\mathcal{T}}$ the sets of feature similarities for all pairs of images in $\mathrm{D}^{\mathcal{S}}$ and $\mathrm{D}^{\mathcal{T}}$, respectively.
Train E to minimize $\mathrm{L}_{init} \leftarrow \mathrm{L}^{\mathcal{S}}_{\mathrm{ID}} + \mathrm{L}_{align}(\mathrm{S}^{\mathcal{S}}, \mathrm{S}^{\mathcal{T}})$.
**PSEUDO-LABELING TRAINING:**
**for** $t = 1$ to $\mathrm{N}_{epoch}$ **do**
**FEATURE EXTRACTION:** Compute target training features $\mathrm{F}^{\mathcal{T}}$ and source validation features $\mathrm{F}^{\mathcal{S}}_{val}$ from $\mathrm{D}^{\mathcal{T}}$ and $\mathrm{D}^{\mathcal{S}}_{val}$.
**AUTO HP-TUNING:** Find $\lambda^*$ that maximizes $\mathscr{L}$ with pseudo-labeling of $\mathrm{F}^{\mathcal{S}}_{val}$ by $\mathrm{C}_{\lambda^*}$ and $\mathrm{D}^{\mathcal{S}}_{val}$ ground-truth labels.
**PSEUDO-LABELING:** Pseudo-label some/all target samples by $\mathrm{C}_{\lambda^*}$ with $\mathrm{F}^{\mathcal{T}}$.
**TRAINING:**
Compute $\mathrm{S}^{\mathcal{S}}_+/\mathrm{S}^{\mathcal{S}}_-$, $\mathrm{S}^{\mathcal{T}}_+/\mathrm{S}^{\mathcal{T}}_-$ the positive/negative sets of feature similarities in $\mathrm{D}^{\mathcal{S}}$ and $\mathrm{D}^{\mathcal{T}}$, respectively.
Train E to minimize $\mathrm{L}_{total} \leftarrow \mathrm{L}^{\mathcal{T}}_{\mathrm{ID}} + \mathrm{L}^{\mathcal{S}}_{\mathrm{ID}} + \mathrm{L}_{align}(\mathrm{S}^{\mathcal{S}}_+, \mathrm{S}^{\mathcal{T}}_+) + \mathrm{L}_{align}(\mathrm{S}^{\mathcal{S}}_-, \mathrm{S}^{\mathcal{T}}_-)$ with $\mathrm{D}^{\mathcal{S}}$ and pseudo-labeled $\mathrm{D}^{\mathcal{T}}$.
**end for**
Return E

**Algorithm 4** AUTO HP-TUNING

**Input:** Number of HP values to validate $N_{search}$
**Input:** Hyperparameter (HP) search function $search\_next()$
**Input:** Source validation set features $F^{\mathcal{S}}_{val}$ and labels $Y^{\mathcal{S}}_{val}$
**Input:** Pseudo-labeling function $C_{\lambda^*}$
**Input:** Pseudo-labeling quality metric $\mathscr{L}$
  Initialize best HP value $\lambda^*$
  Initialize best metric value $L^* \leftarrow -\infty$
  **for** $t = 1$ to $N_{search}$ **do**
    $\lambda \leftarrow search\_next()$
    Get pseudo-labels $\hat{Y}^{\mathcal{S}}_{val}$ by clustering $F^{\mathcal{S}}_{val}$ with $C_\lambda$
    Compute $L \leftarrow \mathscr{L}(\hat{Y}^{\mathcal{S}}_{val}, Y^{\mathcal{S}}_{val})$
    **if** $L \geq L^*$ **then**
      $\lambda^* \leftarrow \lambda$
      $L^* \leftarrow L$
    **end if**
  **end for**
  Return $\lambda^*$

## 5.5 Experiments

### 5.5.1 Datasets and Protocol

#### Datasets

We study HyPASS the same cross-domain Person and Vehicle re-ID adaptation tasks presented in Chapter 4 in Sec. 4.5.4.

Table 5.1: Dataset composition

| Dataset | train IDs | train images | test IDs | gallery images | query images | query images per ID | train images per ID |
|---|---|---|---|---|---|---|---|
| Market [172] | 751 | 12,936 | 750 | 16,364 | 3,368 | 4 | 17 |
| Duke [111] | 702 | 16,522 | 702 | 16,364 | 2,228 | 3 | 24 |
| PersonX [123] | 410 | 9,840 | 856 | 17,661 | 30,816 | 36 | 24 |
| MSMT [140] | 1,041 | 32,621 | 3,060 | 82,161 | 11,659 | 4 | 31 |
| Vehicle-ID [82] | 13,164 | 113,346 | 800 | 7,332 | 6,532 | 8 | 9 |
| Veri [86] | 575 | 37,746 | 200 | 49,325 | 1,678 | 8 | 66 |
| VehicleX [86] | 1,362 | 192,150 | N.A. | N.A. | N.A. | 4 | 141 |

#### Protocol

The feature encoder E is evaluated on the target test set. When it is available, we use the source query set as source validation set $D^{\mathscr{S}}_{val}$ since it is never used elsewhere during the training and no official validation set has been built for these benchmarks. As no test sample is available for VehicleX, we randomly remove 5000 images from the training set to build the validation set. We report the Mean Average Precision (mAP) and rank-1 (top-1) in percents on the target test set after UDA training.

#### Remarks

In the different cross-domain benchmarks, the source validation sets are very varied in size (number of images) and distinct from the target training set in terms of number of IDs and number of samples per ID. According to our theoretical insights in Sec. 5.3.5, we do not expect these statistic differences to influence a good selection of $\lambda$. This will be confirmed by the experiments in Sec. 5.6.3 for further discussion and experiments about this point and the choice of validation set.

### 5.5.2 Implementation Choices and Details

#### Implementation Choices

**Frameworks.** In order to show its effectiveness, we integrate HyPASS within 3 state-of-the-art methods: UDAP [120], MMT [41] and SpCL [42]. We recall that UDAP is a classical pseudo-labeling method, while MMT and SpCL, which manage noise in pseudo-labels, are the best approaches on UDA re-ID. As in the previous chapters, we focus our experiments on these three frameworks for mainly three reasons: these are renowned re-ID approaches, supplied with a code for reproducibility, and with the best UDA re-ID performance on different adaptation tasks (for SpCL particularly).

**Clustering algorithm.** We focus our experiments on DBSCAN [32] clustering for two reasons: it is the most widespread in the state of the art and it is used by the best approaches (cf. Sec. 5.2). Thus, our experiments focus on the selection of HP

$\epsilon$ that is critical for performance (cf. Sec. 5.1). However, experiments are also made with other clustering algorithms (k-means, Agglomerative Clustering [7], HDBSCAN [95]) to show the genericity of HyPASS (cf. Sec. 5.6.3). The main implementation choices are summarized in Tab. 5.2.

Table 5.2: Main implementation choices for experiments.

| Theory | Implementation choices |
|---|---|
| $\lambda$ | Maximum Neighborhood Distance $\epsilon$ |
| $\Lambda$ | Bayesian Search [5] with $\epsilon \in [0,2]$ |
| $C_\lambda$ | DBSCAN [32] |
| $\mathscr{L}$ | Adjusted Random Index (ARI) [110] |
| E | ResNet-50 [51] initialized on ImageNet [25] |
| $L_{align}$ | Maximum Mean Discrepancy (MMD) [113] |
| $s$ | based on $L^2$ distance with normalized features |
| $L_{ID}^{\mathscr{S}}, L_{ID}^{\mathscr{T}}$ | Cross-Entropy & Triplet Losses (UDAP [120] & MMT [41]) |
| | Contrastive Loss (SpCL [42]) |

**Empirical setting comparison.** Pseudo-labeling state-of-the-art approaches use empirical values to set HP $\epsilon$ in DBSCAN. The empirical setting strategy supposes that, in addition to a source labeled dataset, we have access to labels of a part of a calibration target dataset. Therefore, it becomes possible to evaluate the re-ID performance for this cross-dataset adaptation task for different values of $\epsilon$. Then, the $\epsilon$ associated to the best mAP is selected, and reused for other cross-dataset adaptation tasks with another target (unlabeled) dataset.

We can choose PersonX as the source dataset. Indeed, PersonX being a synthetic dataset, it is free to label and it does not raise any problem of privacy access to real people identities. For the sake of a robust empirical setting, we suppose that we have access to the test set of MSMT, the biggest and most challenging person re-ID dataset. We train different models with the best state-of-the-art method SpCL, for different values of $\epsilon$ ($\epsilon = 0.3, 0.4, 0.5, 0.6, 0.7$ see Fig. 5.1), for the cross-dataset adaptation task PersonX→MSMT. The mAP of each model is computed on MSMT test set, and the $\epsilon$ associated with the best mAP is kept. After experiments, as shown on Fig. 5.1, we obtain $\epsilon = 0.6$. This value will therefore be reused for other cross-dataset adaptation task, with other target domains, such as PersonX→Market. In Sec. 5.6.3, we compare HyPASS to this empirical setting strategy (i.e. re-use $\epsilon = 0.6$). Sec. 5.6.1 gives extensive results for more cross-dataset experiments comparing this empirical

strategy with HyPASS.

**HDBSCAN comparison.** HDBSCAN is a hierarchical clustering version of DBSCAN that automatically selects a parameter like $\epsilon$, according to an unsupervised criterion of stability of the clusters in the hierarchy [95]. It therefore seems like a reasonable alternative to DSBCAN with empirical setting since it has an unsupervised heuristic to automatically select an $\epsilon$ value. Indeed, we can see HDBSCAN as an automatic HP tuning of $\epsilon$ and it is therefore relevant to compare HyPASS (DBSCAN) to HDBSCAN on different state-of-the-art methods. The comparison is done in Sec. 5.6.1. HDBSCAN still needs a value for $n_{min}$ controlling the minimum of samples per cluster that is set to 10 during experiments since it gives the best results for different cross-dataset benchmarks in other state-of-the-art work [159].

### Implementation Details

**Data preprocessing.** We build two mini-batches: one of size 64 for source images and another of the same size for target ones. Each batch is made of P=16 identities and K=4 instances per identity (and sampled randomly at initialization phase for target due to lack of labels). Images are resized to 256x128 for person images as in [172, 111, 140] and 224x224 for vehicle ones as in [82, 86]. We randomly flipped and cropped images but we do not use random erasing augmentation during initialization phase since it has been shown to be harmful for direct transfer [92].

**Feature Encoder.** For state-of-the-art comparison, we use a Resnet-50 [51] pretrained on ImageNet [25] as our backbone. The last stride of ResNet-50 is set to 2 to have higher resolution feature map. After the global average pooling layer, we add a BatchNorm layer and then the classification layer(s) which is initialized with the Kaiming initialization [51]. At test time, we use the normalized 2048 pre-classification features with squared Euclidean distance to compute the ranking lists.

**Domain Alignment.** For $L_{align}$, we use the MMD PyTorch implementation of D-MMD paper [96] with the Gaussian kernel [1]. The features are normalized before computing the (conditional) pairwise feature similarities.

[1] https://github.com/djidje/D-MMD

**Initial phase.** The network is trained during 60 epochs. The learning rate is set to $3,5.10^{-4}$ and is decayed by a factor 10 every 20 epochs. Since we have not yet pseudo-labels for the target data, the classical Cross Entropy Loss and Triplet Loss are optimized on the source samples only, jointly with $L_{align}$ on the source and target unlabaled samples.

**HP tuning.** We perform HP search with Bayesian optimization. We choose Bayesian optimization since it is a powerful HP search approach that is able to look for relevant HP values ($\Lambda$) according to an updated belief [5]. We use the library GPyOpt [2] using Gaussian processes. We just used the default Bayesian optimizer parameters using basic Gaussian processes as the modeling function and Expected Improvement (EI) as the acquisition type. The search range for $\epsilon$ is set to $[0,2]$ (it is the whole range of variation for $\epsilon$ since the features are normalized and thus belong to the unit hypersphere). For k-means variant, $k$ is searched in the full range [1, number of target training samples]. At each Auto HP tuning step, we evaluate $N_{HP} = 50$ HP values proposed by the Bayesian search. With this setting, the initial value can be sampled randomly since it has no influence on performance as shown later in Sec. 5.6.3.

The Adjusted Random Index (ARI) [58] is computed between the source validation set ground truth labels and the cluster predictions using the scikit-learn implementation [3].

**Pseudo-labeling training phase.** Implementation details for this step are framework-specific. We put the symbol "*" after the name of the framework to indicate that it corresponds to our version (to include HyPASS and allow easier experimental comparisons) based on the original framework. We give the specific implementation details below. If not specified we make the same choices (optimizer, number of epochs,...) as given in their respective paper.

#### Framework-specific details

**UDAP*.** We build our code from the UDAP [120] implementation publicly available on the official UDAP GitHub [4]. For UDAP, we use an initialization phase be-

[2] https://sheffieldml.github.io/GPyOpt/
[3] https://scikit-learn.org/
[4] https://github.com/LcDog/DomainAdaptiveReID

fore the pseudo-labeling UDA learning. DBSCAN is run on k-reciprocal encoded features with $k = 30$ whereas the k-means version directly uses the feature as in the original paper. The minimum samples $n_{min}$ per cluster is set to 4 (as in paper [120]). Compared to the UDAP paper, we use only one 2048 feature space with Triplet Loss, and add a Cross Entropy Classification loss for the target pseudo-labeled samples (since it improves performance). To add HyPASS, we add to this UDAP* loss, the classification and triplet losses $L_{ID}^{\mathscr{S}}$ for the source samples (by initializing a new classification layer for source IDs) as well as $L_{align}$. Other training hyparameters are the same as in the UDAP paper [120].

**MMT*.** We build our code from the MMT [41] implementation publicly available on the official MMT GitHub [5]. For MMT, we use an initialization phase before the pseudo-labeling UDA learning. DBSCAN is run on k-reciprocal encoded features with $k = 30$ whereas the k-means version directly uses the features as in the original paper [120]. The minimum samples $n_{min}$ per cluster is set to 4. To add HyPASS, we only add to the original MMT global loss function, the hard classification and triplet losses defined in paper [120], for the source samples (by initializing a new classification layer for source IDs), as well as $L_{align}$. Other training hyparameters are the same as in MMT paper [120].

**SpCL*.** We build our code from the SpCL [42] implementation publicly available on the official SpCL GitHub [6]. It does not need an initialization phase and the ID loss on source samples is already implemented and used in the original framework with the contrastive loss. To include HyPASS, we add $L_{align}$ to the global objective and remove the cluster criterion (for HyPASS and HDBSCAN experiments). Other hyparameters are the same as in the SpCL paper [42].

Our implementations based on the authors' code for UDAP*, MMT* gives better performance than those reported in the papers. For SpCL*, we obtained only slightly inferior performance (-1.1 p.p. at worst), which should not interfere with conclusions that will be made from experiments in Sec. 5.6.

[5] https://github.com/yxgeee/MMT
[6] https://github.com/yxgeee/SpCL

Table 5.3: Comparison of HyPASS with empirical setting strategy on pseudo-labeling state-of-the-art methods on person re-ID adaptation tasks. * means we used authors' code and add HyPASS.

| Method | HP selection | Market→MSMT | | PersonX→Market | | PersonX→MSMT | | Market→Duke | | Duke→Market | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 |
| UDAP* [120] | Empirical ($\epsilon = 0.6$) | 12.0 | 30.6 | 48.4 | 68.4 | 10.5 | 26.3 | 50.1 | 70.2 | 55.3 | 78.1 |
| | HDBSCAN | 11.8 | 29.8 | 48.1 | 68.3 | 10.3 | 25.9 | 51.3 | 72.5 | 55.9 | 80.0 |
| | **HyPASS** | **21.4** | **48.8** | **62.2** | **73.7** | **15.6** | **36.4** | **64.9** | **78.0** | **69.8** | **87.1** |
| MMT* [41] | Empirical ($\epsilon = 0.6$) | 23.8 | 49.9 | 71.1 | 66.8 | 17.4 | 39.0 | 65.3 | 78.1 | 73.6 | 89.4 |
| | HDBSCAN | 23.0 | 47.8 | 70.9 | 66.1 | 18.0 | 41.1 | 65.2 | 78.2 | 74.2 | 90.1 |
| | **HyPASS** | **25.1** | **52.2** | **74.5** | **88.9** | **20.3** | **45.9** | **68.8** | **82.8** | **76.0** | **90.4** |
| SpCL* [42] | Empirical ($\epsilon = 0.6$) | 25.7 | 53.4 | 72.2 | 86.1 | 22.1 | 47.7 | 68.3 | 82.5 | 76.1 | 89.8 |
| | HDBSCAN | 24.6 | 52.0 | 70.8 | 86.5 | 21.1 | 46.9 | 66.4 | 81.3 | 75.8 | 89.5 |
| | **HyPASS** | **27.4** | **55.0** | **77.9** | **91.5** | **23.7** | **48.6** | **71.1** | **84.5** | **78.9** | **92.1** |

Table 5.4: Comparison of HyPASS with empirical setting strategy on pseudo-labeling state-of-the-art methods on vehicle re-ID adaptation tasks. * means we used authors' code and add HyPASS.

| Method | HP selection | VehicleID→Veri | | VehicleX→Veri | |
|---|---|---|---|---|---|
| | | mAP | rank-1 | mAP | rank-1 |
| UDAP* [120] | Empirical ($\epsilon = 0.6$) | 35.6 | 74.1 | 35.0 | 75.9 |
| | HDBSCAN | 35.9 | 75.0 | 35.5 | 79.9 |
| | **HyPASS** | **36.9** | **74.9** | **37.0** | **77.0** |
| MMT* [41] | Empirical ($\epsilon = 0.6$) | 36.4 | 74.2 | 36.3 | 75.8 |
| | HDBSCAN | 37.0 | 75.9 | 36.5 | 75.9 |
| | **HyPASS** | **36.9** | **75.0** | **36.8** | **76.1** |
| SpCL* [42] | Empirical ($\epsilon = 0.6$) | 37.6 | 79.7 | 37.4 | 81.0 |
| | HDBSCAN | 37.4 | 79.9 | 37.5 | 79.8 |
| | **HyPASS** | **40.0** | **81.1** | **40.3** | **81.9** |

## 5.6 Results and analysis of HyPASS.

### 5.6.1 Effectiveness of HyPASS on state-of-the-art methods.

**Performance analysis of HyPASS.**

**HyPASS vs empirical setting.** Results in resp. Tab. 5.3 and Tab. 5.4 show that our automatic HP selection improves the three state-of-the-art frameworks, on all person re-ID and vehicle re-ID adaptation tasks. This improvement is particularly significant for UDAP: it increases, e.g., the mAP by +9.4 p.p. on Market→MSMT and +13.8 p.p. on PersonX→Market over the empirical setting strategy. This improvement of using HyPASS over the empirical setting strategy is also consistent for "eas-

ier" adaptation tasks such as Duke→Market (+14.5 p.p.) and Market→Duke (+14.8 p.p.). HyPASS seems thus to benefit a simple pseudo-labeling approach like UDAP by making it competitive with more complex approaches like MMT, designed to be resistant to pseudo-label noise. Our contribution also improves consistently MMT and SpCL (the best state-of-the-art approaches) on all tasks: there is, e.g., up to +4.1 p.p. mAP improvement on PersonX→Market for SpCL compared to the SpCL reported performance (using empirical setting). Furthermore, we highlight that SpCL with HyPASS for cross-dataset UDA re-ID is able to outperform (or at least be competitive with) performance of the latest UDA re-ID and unsupervised approaches: for example, SpCL + HyPASS reaches 71.1 % mAP on Market→Duke whereas [142, 144, 160, 167, 16, 166] reach respectively, 59.1%, 53.8%, 69.2%, 69.2%, 69.1% and 67.6% mAP on Duke or Market→Duke.

We recall that experiments have been conducted with an empirical setting performed on PersonX→MSMT ($\epsilon = 0.6$). A different empirical setting choice, on PersonX→Market for example, would let to an empirical value $\epsilon = 0.5$ (see Fig. 5.1), and therefore improvements given by using HyPASS would be greater on other cross-datasets. Indeed, with $\epsilon = 0.5$, the performance are further degraded for SpCL on PersonX→MSMT (20.3% mAP). Therefore HyPASS improves the mAP by +3.4 p.p with this other empirical setting for SpCL.

**HyPASS vs HDBSCAN.** Moreover, results in Tab. 5.3 and Tab. 5.4 show that using HyPASS (with DBSCAN) consistently outperforms HDBSCAN for the three frameworks and on all the person & vehicle re-ID cross-datasets benchmarks. Indeed, results show that HDBSCAN is in fact not necessarily better than using the empirical setting $\epsilon = 0.6$ (for e.g. 24.6% mAP for SpCL on PersonX→Market with HDBSCAN instead of 25.7% mAP with empirical setting) or only brings small improvements (+0.1 p.p. for SpCL and PersonX→Market with HDBSCAN instead of empirical setting). Therefore, the conclusions done for empircal setting vs HyPASS remains the same for emprical setting vs HDBSCAN: among those three HP selection strategies, using HyPASS appears to be the best one.

### 5.6.2 A cluster quality analysis to understand the effectiveness of HyPASS.

To understand more precisely the positive impact of HyPASS on the training process, we monitor the evolution of: (i) the quality of the clusters found during

the pseudo-labeling cycles, through the ARI of the pseudo-labeled target samples, every 10 epochs (after the pseudo-labels are updated); (ii) HP $\epsilon$ found by HyPASS. Fig. 5.3 shows that HyPASS seems to find better clusters (with better ARI) than the fixed empirical parameter strategy ($\epsilon = 0.6$) from the first epochs on. We believe this impact on the quality of the clusters is 'iterative': better clusters (pseudo-labels) in early epochs will imply the learning of better representations and therefore the possibility to make better clusters when the pseudo-labels are updated. Fig. 5.3 also highlights that the value of the selected $\epsilon$ changes cyclically (as the feature representation changes) over the pseudo-labeling cycles.

### 5.6.3 Ablative Study & Parameter Analysis on training time and performance.

#### Relevance of the optimization losses

In the ablative study presented in Tab. 5.5, we seek to verify the relevance of our optimization losses (see Eq. 5.14) for the selection of HP for the UDAP [120], MMT [41] and SpCL [42] approaches.

We train different variants by removing terms from the total loss function (cf

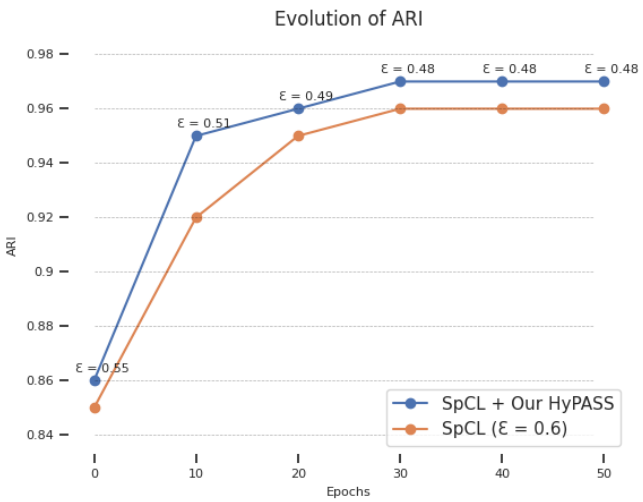


Figure 5.3: Positive impact of an iterative HP tuning of $\epsilon$ (HyPASS) on the clustering quality. The figure represents evolution of ARI of the pseudo-labeled target training set through epochs on PersonX→Market with SpCL [42] each 10 epochs. Above each point is indicated the value of $\epsilon$ automatically selected by HyPASS.

Eq. 5.14) in order to observe their effects on the final performance (mAP). Variant #5 corresponds to HyPASS with the total loss function. First, we notice that training the model to be discriminating on the source domain (variant #3) together with our Auto HP tuning allows improvements compared to variant #1 (only Auto HP tuning) for UDAP: +18.6 p.p. mAP on PersonX→Market. We believe that the feature encoder in variant #1 specializes on target domain while forgetting source domain initialization. Thus, HP selection becomes worse because it is done on a representation that is less and less discriminating for the source domain over time. After a certain number of epochs, bad choices of HP may impact the quality of pseudo-labels and, then, target representation. In variant #2, performance drops even more if alignment is added without $L_{ID}^{\mathcal{S}}$ (variant #2): -28.7 p.p. on PersonX→Market . We believe that alignment on poorly discriminative source is even more harmful to the target representation. We notice the same behavior for MMT with -29.4 p.p. and -8.7 p.p. respectively. Therefore, when using Auto HP of HyPASS, it is necessary to keep optimizing source ID-discriminative features with $L_{ID}^{\mathcal{S}}$.

Adding the term $L_{align}^{cond}$ of conditional domain alignment of feature similarities (variant #5) further improves substantially performance by using Auto HP (variant #3): +13.4 p.p. on PersonX→Market . The same improvement trend is observed for MMT and SpCL. This seems to confirm our theoretical considerations of reducing the variance of the estimation by reducing the domain discrepancy in the feature similarity space when using Auto HP (see Sec. 5.3.5).

Finally, by comparing variants #4 and #5, we observe the contribution of our cyclic Auto HP: +9 p.p. on PersonX→Market . The same is true for MMT and SpCL. We believe this shows the importance of choosing a suitable HP for each pseudo-labeling update cycle as done with the Auto HP tuning step of HyPASS (variant #5).

**Performance of HyPASS with other clustering algorithms.**

**K-means.** Other clustering algorithms can be used instead of DBSCAN. But they still need to set HP. For example, k-means relies on the number $k$ of clusters. Similarly to the sensibility of DBSCAN with $\epsilon$, Fig. 5.4 shows that the performance with k-means is also sensible to the number of clusters HP. Again, choosing a good HP value is crucial to get good performance: for example, by choosing k = 250, performance drops from 70.8% to 50.2% mAP for PersonX→Market and from 16.6% to 10.1% compared to k = 500 for PersonX→MSMT with MMT .
Therefore, empirical setting strategy for choosing k on another adaptation task quite

Table 5.5: Ablation studies on HyPASS for UDAP*, MMT* and SpCL* methods (mAP in %). #5 is (full) HyPASS.

| Method | # | Losses | | | Auto. HP tuning | PersonX →Market |
|---|---|---|---|---|---|---|
| | | $L_{ID}^{\mathcal{T}}$ | $L_{ID}^{\mathcal{S}}$ | $L_{align}^{cond}$ | | mAP |
| UDAP* [120] | 1 | ✓ | | | ✓ | 30,2 |
| | 2 | ✓ | | ✓ | ✓ | 20.1 |
| | 3 | ✓ | ✓ | | ✓ | 48.8 |
| | 4 | ✓ | ✓ | ✓ | | 53.2 |
| | 5 | ✓ | ✓ | ✓ | ✓ | **62.2** |
| MMT* [41] | 1 | ✓ | | | ✓ | 55.9 |
| | 2 | ✓ | | ✓ | ✓ | 41.3 |
| | 3 | ✓ | ✓ | | ✓ | 70.7 |
| | 4 | ✓ | ✓ | ✓ | | 71.5 |
| | 5 | ✓ | ✓ | ✓ | ✓ | **74.5** |
| SpCL* [42] | 3 | ✓ | ✓ | | ✓ | 68.1 |
| | 4 | ✓ | ✓ | ✓ | | 73.9 |
| | 5 | ✓ | ✓ | ✓ | ✓ | **77.9** |

limits performance too. Indeed, re-using the best value from PersonX→MSMT (k=1500) leads to 59.8% mAP for PersonX→Market whereas it could have been 70.8% for k=500. Reciprocally, choosing k=500 from PersonX→Market leads to 13.6% for PersonX→MSMT instead of 17.4% for k=1500.

As illustrated on Fig. 5.4 and shown in Tab. 5.6, using HyPASS leads to better performance compared to empirical setting. For PersonX→Market with MMT, it leads to 71.1% mAP instead of 59.8% reusing k=1500 obtained by empirical setting on PersonX→MSMT.

**Agglomerative Clustering.** Agglomerative Clustering [7] is another clustering algorithm that can be used instead of DBSCAN. As DBSCAN, Agglomerative Clustering is a density-based clustering algorithm that relies on a neighborhood distance threshold parameter $\epsilon$. HyPASS can also improve performance of pseudo-labeling methods using this clustering algorithm. As shown in Tab. 5.6, for PersonX→Market with SpCL, using HyPASS leads to 78.2% mAP instead of 72.2% using empirical setting ($\epsilon = 0.6$).

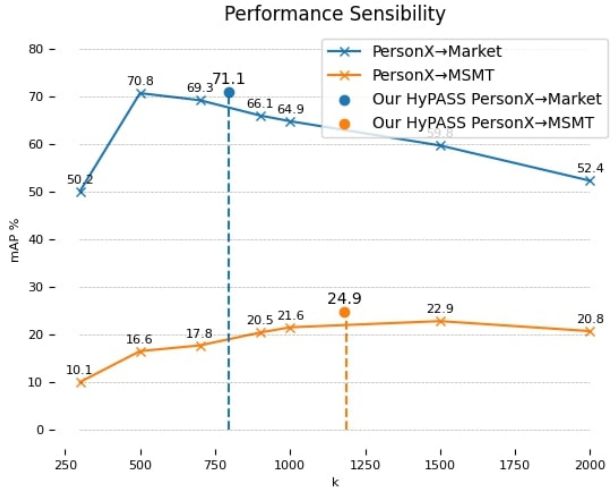


Figure 5.4: Performance sensibility for the state-of-the-art framework MMT [41] with respect to $k$ parameter of k-means. HyPASS performs cyclic pseudo-labeling HP tuning and for more clarity we only represent in the Figure the final performance for $\epsilon$ value found for the last training stage.

Table 5.6: Performance (mAP) of HyPASS on k-means and Agglomerative Clustering and with state-of-the-art pseudo-labeling approaches. We set $k = 1500$ as empirical setting since it is the best configuration on PersonX→MSMT in our experiments. For Agglomerative Clustering, empirical setting $\epsilon = 0.6$ is motivated by analogy to our experiments for PersonX→MSMT with DBSCAN (see Fig. 5.1) which is also a density-based algorithm.

| Method | Clustering | HP choice | PersonX→Market |
|---|---|---|---|
| MMT* [41] | k-means | Empirical $k = 1500$ | 59.8 |
| | | **HyPASS** | **71.1** |
| SpCL* [42] | Agglo. Clustering [7] | Empirical $\epsilon = 0.6$ | 72.6 |
| | | **HyPASS** | **78.2** |

**Influence of the validation set size.**

We have seen that HyPASS brings consistent improvements on various adaptation tasks (see Tab. 5.3) and therefore with various sizes validation set (see Tab. 5.1 number of query validation images). These also show experimentally that performance improvements from HyPASS is also robust to various dataset compositional bias between the source and target domains, more particularly the difference in number of query per IDs and IDs.
But the validation set size also intuitively influences the clustering computation

Table 5.7: Experiments with different validation set size $N^S_{val}$ on SpCL for PersonX→Market showing the validation set size on performance and training computation time.

| | Empirical setting | HyPASS $N^S_{val}$ = 1000 | 5000 | 10000 | 30816 |
|---|---|---|---|---|---|
| Time | 60h12 (6 × ∼ 10h02) | 12h08 | 34h39 | 42h21 | 68h43 |
| mAP (in %) | 72.2 | 76.1 | 77.8 | 77.8 | 77.9 |

time, and thus the full training computation time of the frameworks where HyPASS is added. Moreover, it is also interesting to have more experimental insights on the influence of the validation set on performance improvement of HyPASS for a fixed adaptation task. That's why we further investigate the influence of the validation set size on the training computation time and the re-ID performance. Experiments are conducted on PersonX→Market for the SpCL framework. For this, we randomly select N images from PersonX query set. The execution time (on the same machine) and re-ID performance are reported in Tab. 5.7. The empirical setting strategy has been performed on PersonX→Market adaptation task with the 5 HP values: $\epsilon = 0.3, 0.4, 0.5, 0.6, 0.7$. The empirical setting strategy requires 5 training of SpCL with the 5 HP values for PersonX→MSMT then one more training of SpCL for PersonX→Market with the best $\epsilon$ ($\epsilon = 0.6$) evaluated by the mAP on the target test set.

We notice that the training computation time increases with the validation set size. However, it is still fairly reasonable for a training time including HP selection. Even with a large validation set (30k images), the complete training time lasts only 68h40 and brings significant performance for this adaptation task (+5.7 p.p.). In practice, it is quite big for a validation set size, and experiments show that even with 5k images, performance remains the same, with a training computation time reduced by about 25h33 compared to empirical setting. More generally, performance of HyPASS is not really sensible validation set size variations tested (from 1/10 up to 3 times the size of the training set induced only 1.8 p.p. variation). Indeed, this is consistent with our guess that reducing the domain discrepancy should allow less sensibility to the number of validation samples, as motivated by Eq. 5.7 in Sec. 5.3.4.

**Influence of Auto HP selection criterion.**

We included in the design of HyPASS different modeling choices aiming at improving training time and performance. To show the relevance of these choices, we conducted various experiments by changing HyPASS HP selection strategy on PersonX→MSMT on the framework SpCL. First, HyPASS HP selection is directly based on cyclic clustering quality evaluations instead of re-ID performance evaluation in order to reduce the computation cost. As illustrated in Tab. 5.8, using HyPASS but with the mAP criterion (the re-ID criterion which is our main task) on the source test to select the clustering HP gives almost the same performance of HyPASS (77.1% mAP), but greatly increases the training time to 90h29 instead of 68h43. We reckon that it is mainly due to the higher number of training steps needed to evaluate HP values with the mAP. Even though the best target mAP is the final goal, our assumption to select HP by clustering quality evaluation instead of mAP evaluation (Sec. 5.3.1) is relevant to limit the training time while having the best re-ID performance.

Table 5.8: Impact of HyPASS with different version of Auto HP criterion on the re-ID performance and computation. Experiments done on SpCL for PersonX→Market.

| Variants | Auto HP criterion | mAP | Time |
|---|---|---|---|
| SpCL w/ mAP HP selection | re-ID task (mAP) | 77.1 | 90h29 |
| SpCL w/ HyPASS | clustering task (ARI) | 77.9 | 68h43 |

**Performance with other HyPASS variants**

**Domain Alignment.** We conducted some experiments with other implementation choices for HyPASS with SpCL on PersonX→Market. For instance, a 2-layer Domain Adversarial Neural Network (DANN [39]) can be used instead of MMD to align the pairwise feature similarities. Tab. 5.9 shows that HyPASS keeps performance improvement over the framework without HyPASS (+5.1 p.p. mAP compared to SpCL without HyPASS).

**Cluster quality criterion.** The Normalized Mutual Information (NMI) can replace the ARI and gives as good performance (+0.2 p.p. mAP in Tab. 5.9 compared to HyPASS with ARI).

**HP search strategy.** Using a more simple HP search strategy like grid search for $\epsilon \in \Lambda = [0.05, 0.1, 0.15, ..., 2]$ can replace the Bayesian search. It still gives good results with HyPASS (-0.3 p.p. compared to Bayesian search in Tab. 5.9).

**HP search initialization.** In Tab. 5.10, when using Bayesian Search with $N_{HP} = 50$ proposed values per HP tuning phase, the initial value $\epsilon_0$ has completely no impact on performance.

Table 5.9: Performance of HyPASS for PersonX→Market with SpCL* [42] pseudo-labeling method on different variants.

| Method | PersonX→Market |
|---|---|
| SpCL* | 72.2 |
| SpCL*+ HyPASS (MMD + Bay. search + ARI) | 77.9 |
| SpCL* + HyPASS (DANN [39] + Bay. search + ARI) | 77.3 |
| SpCL* + HyPASS (MMD + Bay. search + NMI) | 78.1 |
| SpCL* + HyPASS (MMD + Grid Search + ARI) | 77.6 |

Table 5.10: Robustness of HyPASS against the Bayesian search initialization of $\epsilon_0$. Performance (mAP in %) for PersonX→Market of HyPASS with SpCL* [42] pseudo-labeling method are reported with different values of Bayesian Search initialization.

| Bayesian search initialization $\epsilon_0$ | PersonX→Market |
|---|---|
| 0.01 | 77.8 |
| 0.8 | 77.9 |
| 2 | 77.8 |

### 5.6.4 Extension to an industrial use case of cattle re-ID

HyPASS has demonstrated its effectiveness on multiple domain adaptation tasks from academic benchmarks. In order to confront our work to an industrial case, it has been extended in the framework of an industrial partnership between the CEA and the start-up AIHerd. This extended work is more particularly issued from a joint work with Lapouge Guillaume and Luvison Bertrand, researchers at CEA. As AIHerd works on cow-monitoring application, the industrial use case tackle the problem of cow re-ID. This collaboration for cow re-ID has been an opportunity to successfully apply and adapt HyPASS to a different industrial use case, keeping in mind the

cross-domain challenge and the deployability of the model. For more details about this additional collaboration, the reader can refer to the appendix (Chapter A).

## 5.7 Conclusion and discussion

This chapter addresses a problem of deployability, concerning the HP selection for pseudo-labeling UDA re-ID approaches, as it can have a negative impact on performance when addressing new unlabeled target datasets. This chapter provided novel theoretical insights to highlight the conditions under which a source-based HP selection by model selection is effective for the UDA clustering task. These insights allowed to design a new general method, HyPASS, which automatically selects suitable clustering HP of pseudo-labeling UDA methods. It is based on Source-Guidance and domain similarity alignment. When HyPASS is applied to select critical clustering HP, instead of using empirical settings, it consistently improves performance of the best state-of-the-art methods for person and vehicle re-ID. While HyPASS is a solution to improve the deployability of pseudo-labeling UDA methods, Unsupervised re-ID state-of-the-art methods are also pseudo-labeling methods based on clustering. Therefore, these methods should also be confronted with the lack of deployability due to the choice of clustering HP. However, HyPASS cannot be applied as it is designed for Unsupervised re-ID, since it relies on Source-Guidance, and therefore need source labeled data. A future research direction could be to design an effective clustering HP strategy that could extend to Unsupervised re-ID setting.

# Chapter 6

# Conclusion and perspectives

As motivated in Chapter 1, to address the practical problem of cross-domain re-ID performance drop, without increasing the annotation costs, this thesis investigates the learning framework of Unsupervised Domain Adaptation (UDA). UDA re-ID aims at leveraging the knowledge from a labeled source dataset and an unlabeled target dataset, to train a re-ID feature encoder that maximizes its re-ID performance on the target domain. The bibliography analysis in Chapter 2 leads this work to more particularly focus on the promising pseudo-labeling UDA re-ID methods. To improve them, the general idea is to leverage in an efficient way, the useful re-ID knowledge in the labeled source data to train pseudo-labeling UDA re-ID methods. In Chapter 3, we empirically investigate the use of the source data, by designing and evaluating different pseudo-labeling methods by including the source data during the training phase. By reducing the source-bias during the training, using some intuition-based good practices implemented during the model training, source-guided pseudo-labeling is able to give some positive results, outperforming target-only pseudo-labeling. Given the initial positive experimental results of this Chapter, indicating that the source knowledge can benefit a pseudo-labeling method, a general theoretical framework for pseudo-labeling UDA re-ID is proposed in Chapter 4. The goal is to derive a set of conditions and good practices to systematically benefit from the source knowledge during training of a pseudo-labeling method. Moreover, this theoretical framework gives us more understanding about how the pseudo-labeling method can be improved by using the source data during training. While Chapters 3 and 4 show that the source knowledge can be leveraged for pseudo-labeling methods during the training phase to improve their cross-domain performance, Chapter 5 proves that the source

knowledge can also be used to enhance the deployment of these methods. In this chapter, it is shown that the source can be used as a way to select hyperparameters values essential to the robustness of the cross-domain re-ID performance of these methods. Overall, the research work carried out during this thesis leads to improved cross-domain re-ID models, enhancing their practical performance evaluated on various cross-domain re-ID benchmarks: pedestrian, vehicle and cattle re-ID (see Appendix A). This empirically suggests a part of robustness of designed methods to tackle some cross-domain re-ID problems regardless of the semantic class of object to re-identify. Moreover, this thesis provides more theoretically-grounded insights to understand the role of the source knowledge to improve pseudo-labeling methods.

While research in UDA successfully improves cross-domain re-ID, other research directions that could tackle cross-domain re-ID have also improved: Unsupervised Learning for re-ID (Unsupervised re-ID) and Domain Generalization for re-ID (Generalizable re-ID). With the aim of identifying perspectives beyond this work, which has been digging in the direction of UDA, it is relevant to review and discuss new advances in Generalizable re-ID and Unsupervised re-ID.

## 6.1 Unsupervised re-ID

Unsupervised re-ID, as pseudo-labeling UDA re-ID, is based on pseudo-labeling [80, 42, 68, 130, 155, 138, 134, 15]. While some methods focus on using the temporal information in the frames to generate the pseudo-labels [68], the majority cluster the training data features as for pseudo-labeling UDA re-ID [80, 42, 130, 155, 138, 134, 15]. Unlike UDA, the lack of source data generally implies initialization of the first pseudo-labels by transfer learning, from a feature encoder (pre-)trained for classification on ImageNet. Using an ImageNet classification feature encoder for re-ID indeed provides very poor re-ID performance (about 2% mAP on Market for a ResNet-50 architecture). Therefore, pseudo-labels initialized from this feature representation are expected to be much noisier than those obtained by a feature encoder optimized for supervised re-ID with a source dataset from another domain. By designing pseudo-labeling UDA re-ID methods with better noisy-label robustness and pseudo-label refinery strategies [42], pseudo-labeling Unsupervised re-ID has jointly improved. Some Self-Supervised Contrastive learning techniques are

also adapted for the re-ID task, such as MoCo [50] which inspires the Hybrid Memory and the contrastive loss of SpCL [42] for Unsupervised re-ID and UDA re-ID. However, it seems impossible to directly apply Self-Supervised Contrastive Learning methods designed for classification to re-ID, which rely on data augmentation of data instances. For example, MoCo directly applied to re-ID gives 6.1% mAP on Market [42]. Unsupervised re-ID seems to still need to cluster the training features to generate the informative positives from a same cluster. This might be explained by the fine-grained nature of the identity information for re-ID compared to the class defined for classification. Therefore, as for UDA re-ID, it can be expected that Unsupervised re-ID remains a specific research topic in parallel to those around classification, due to the specificities of re-ID as a computer vision task.

As seen in Tab. 6.1, currently the performance gap between recent UDA re-ID and Unsupervised re-ID methods is relatively small: 78.9% mAP vs 73.1% mAP resp. for a recent UDA re-ID and Unsupervised re-ID method [42] on the dataset Market. While using a labeled source dataset can significantly improve the performance on the target domain by +5.8 p.p. mAP, Unsupervised re-ID performance is relatively not so far from Supervised re-ID, while being completely label-free.

Table 6.1: Comparison of recent methods designed for UDA re-ID ([42] + HyPASS from Chapter 5), Unsupervised re-ID ([42]) and Generalizable re-ID ([23]). The best cross-dataset benchmarks are reported for each target dataset, for UDA re-ID. Generalizable re-ID uses the other two dataset for training as source datasets. mAP and rank-1 are reported in %.

| Paradigm | Duke | Market | MSMT |
|---|---|---|---|
| | mAP | mAP | mAP |
| UDA re-ID ([42] + HyPASS) | 71.1 | 78.9 | 27.4 |
| Unsupervised re-ID ([42]) | 65.2 | 73.1 | 19.1 |
| Generalizable re-ID ([23]) | 56.9 | 56.5 | 13.5 |
| Supervised re-ID ([42]) | 74.6 | 84.4 | 82.3 |

## 6.2 Generalizable re-ID

Generalizable re-ID aims at learning a feature encoder that generalizes well to an unseen target domain, using only labeled source data. Some methods consider multiple source-domain, and as UDA, use Domain Alignment techniques to learn

a Domain Invariant feature representation [119]. Other methods exploit Few-Shot learning techniques, such as meta-learning [19], in order to learn a generalizable model from a small labeled source dataset. Nevertheless, Generalizable re-ID is significantly outperformed by UDA re-ID: a recent Generalizable re-ID method gives 56.5% mAP on Market [77], vs 26.8% mAP for UDA re-ID [42] (see Tab. 6.1). However, Generalizable re-ID is still a key research direction for cross-domain re-ID, since target data might not be easily accessible due to some practical constraints (e.g.: data privacy reasons). We reckon that future advances in Few-Shot Classification and Domain Alignment for UDA classification further improve Generalizable re-ID.

## 6.3 What could be the best solution for cross-domain re-ID ?

While Generalizable re-ID performance are still far from UDA, given the reduced performance gap between Unsupervised re-ID and UDA re-ID, should future researches for cross-domain re-ID focus only on Unsupervised re-ID ? Indeed, Unsupervised re-ID is in addition the cheapest solution considering the labeling, since no label is required at all.

### 6.3.1 Limits in cross-domain adaptability ?

UDA assumes access to labeled source data and unlabeled target data. Given some practical constraints, we might not easily access these data for some applications, for example for privacy reasons concerning pedestrian re-ID. Contrary to UDA, Generalizable re-ID can allow training with only a set of synthetic labeled data, without the need to use data collected from real people. Unsupervised re-ID can also be more flexible in term of data accessibility, since it does not need manual annotation of any data, that can leak private information.
Moreover, as seen in Tab. 6.1, cross-domain adaptability through UDA significantly outperforms Unsupervised re-ID by +5.8 to +8.3 p.p. mAP. As these two experiments are done using the same pseudo-labeling method, we can assume that these performance improvements come from the source knowledge. But in the same way, we could also state that most of the performance of the UDA re-ID method comes from the target knowledge. This raises the question of how much knowledge can be leverage from a source domain to benefit cross-domain re-ID.

As mentioned in Chapter 2, non-pseudo-labeling approaches build their supervision from the labeled source data. They can hardly compete with pseudo-labeling approaches, which draw their supervision from the target data through predicted pseudo-labels. In these pseudo-labeling approaches, the source domain generally intervenes only to generate the first pseudo-labels. The general direction of this thesis work has been to further rely on the source data to improve pseudo-labeling methods. More particularly, our theoretical framework developed in Chapter 4 highlights can help us to answer the question about the amount of useful source knowledge. It indeed depends on various parameters as the number of source and target data, the class of models considered to learn the feature encoder, the domain discrepancy between the source and target domains, the noise level in pseudo-labels...More particularly, we have noticed from the theory a multiplicative interaction between the good practice deduce from the theory: improving the quality of pseudo-labels or the robustness to noisy pseudo-labels, reduce the performance improvement from using the source data. In this chapter, we have also highlighted experimentally that improving the quality of pseudo-labels and improving the robustness of the model to noisy labels are the good practice that generally bring the highest cross-domain performance gains. Therefore, in order to improve the cross-domain performance, it seems more judicious to focus on these two aspects (noisy pseudo-label robustness and noise reduction in pseudo-labels). As this 2 good practices can be applied in an Unsupervised re-ID pseudo-labeling method, we can expect in the future, by focusing research on these two aspects, that we might reach a point where the source knowledge become useless to cross-dataset benchmarks where it used to improve performance.

However, if we consider the more practical aspect of deployability, Unsupervised re-ID being based on pseudo-labeling, it remains dependent on the clustering hyperparameter values in order to obtain good performances, as detailed in Chapter 5. As for UDA re-ID, no label is available for the target dataset, and therefore classical hyperparameter selection techniques with a target validation set cannot be used. Although a solution to this problem has been proposed by using the source data as a validation set in Chapter 5, it is not applicable at all for Unsupervised re-ID with no labeled source data. Since this problem has not been addressed yet for Unsupervised re-ID, it is likely to suffer from performance reliability problems after deployment. Therefore, for a better deployability in cross-domain settings, using a labeled source dataset for automatic hyperparameter selection can be performed

with UDA re-ID.
Ideally, in order to combine cross-domain performance and deployability, we reckon that Generalizable re-ID, UDA re-ID and Unsupervised re-ID could each contribute to build better pseudo-labeling frameworks, in the following way:

- With Generalizable re-ID, a more generalizable feature encoder can be trained with the labeled source data, thus giving a better initialization for pseudo-labeling.
- With Unsupervised re-ID, the last advances in Self-Supervised Contrastive Learning for Classification could be adapted to re-ID, in addition to techniques improving the quality of the pseudo-labels and the robustness to noisy labels, for a better re-ID model on the target domain.
- With UDA re-ID, the deployability of the method could be improved by choosing automatically hyperparameters for pseudo-labeling.

## 6.4 Bridging Cross-modal & UDA re-ID

This thesis work focused on the problem of cross-domain re-ID performance drop. Another practical problem of re-ID is the use of cross-modal data. This problem arises with the use of different types of information, extracted from different sensors, to perform the re-ID task. For example, we can imagine a re-ID application working in a camera network, composed of RGB cameras in the lit spaces, and infrared cameras otherwise allowing night vision.
The general idea to tackle this cross-modal re-ID problem is learning a feature encoder allowing to compare the features of various modalities to perform re-ID [67, 48, 102]. By considering the modalities as belonging to the same space (e.g.: RGB and infrared images in the image space), and the data coming from different probability distributions on this space, the cross-modal re-ID boils down to a cross-domain re-ID problem, more particularly focusing on Domain Translation. These UDA re-ID approaches therefore become relevant to solve this cross-modal re-ID problem, and thus UDA re-ID can inspire solutions to reduce the gap between modalities (domains) in the input space (with image-to-image translation with style transfer) or features (learning a modality-invariant feature space). Some

cross-modal methods [131] seem to have already been inspired by UDA re-ID, and we reckon that UDA re-ID advances could help to address the challenging but related cross-modal re-ID problem.

# Appendix A

# Appendix

## A.1 An industrial use case of cattle re-ID

### A.1.1 Motivations

Animal monitoring is a great interest for farmers to improve the farm's productivity. Indeed, cow monitoring can provide information in real time about health, nutrition, fertility and location of every cow. In general, re-ID is of great interest for animal monitoring applications based on computer vision [115, 81, 85, 84] in order to track and monitor each cow in the farm. However, re-ID has been mainly focused on people and vehicles. Some work exists for animal re-ID, focusing for instance on Amur tiger re-ID [81, 70, 85]. Concerning cattle re-ID, the specificity of the appearance of cows has led to specific works [9, 1, 3, 2, 40], studying it under the frameworks of supervised learning or self-supervised learning [40] assuming access to tracklets. To the best of our knowledge, cattle re-ID has therefore never focused on the cross-domain re-ID performance drop, which is of great practical interest as for cross-domain person re-ID [148]. However, the classical cross-domain UDA framework has been designed for person and vehicle re-ID, and therefore can be limited to deal with all practical problems posed by cross-domain cattle re-ID. First, classical UDA seeks to maximize performance on the target domain. To achieve this goal, the learned UDA model can specialize on the target domain while forgetting or ignoring the source domain knowledge as seen in Chapter 3. However, in practice, if new cameras are deployed in a farm, we generally want to adapt the re-ID model to them, while preserving good performance on the old ones. We call this property *source conservation*.

Moreover, supposing that we could keep the data collected from multiple farms of interest: could we take advantage of the quantity and diversity of this data, in order to accumulate knowledge, and therefore improve the discriminating power of the model on all domains ? In a cattle re-ID real-world scenario, data can be collected from a set of farms of interest, where each farm defines a target domain. Therefore, it would be of a great practical interest to be able to learn a model that can learn from multiple domains and accumulate all the useful knowledge from them. This kind of UDA framework can be described as *cumulative* and *multi-target*. As classical UDA does not include these specific properties (source conservation, cumulative and multi-target) motivated by cattle re-ID applications, we call it: Cumulative Unsupervised Multi-Domain Adaptation (CUMDA).

Multi-target domain adaptation approaches have been designed to tackle this cross-domain challenge for object classification and semantic segmentation [105, 114, 43, 18]. However, these two tasks consider the same semantic classes for all domains, and thus between training and testing (closed-set). As these methods are designed considering a closed-set setting, they cannot be directly used for a re-ID task, for which the identity classes can differ between training and testing (open-set), and even between the multiple target domains. To our knowledge, multi-target domain adaptation has actually not been addressed for re-ID. Moreover, the Domain Generalization framework [132], which aims at learning a model performing well on unseen target domains, differs from CUMDA. Indeed, CUMDA assumes that all the target domains are known at training time, and that unlabeled training data from them are available. CUMDA therefore aims at specializing the model for each of the target domains while accumulating the knowledge from all of them to improve the cross-domain re-ID performance. Although generalization is not the goal, we expect that a CUMDA model, by accumulating knowledge from multiple target domains, can improve its generalization on new unseen domains.
Beyond just applying HyPASS to an existing UDA framework originally designed for person or vehicle re-ID, for cow re-ID UDA, we propose in this section to design a complete CUMDA framework for cross-domain cow re-ID, that encompasses the specificities of a real-world cattle re-ID application.

## A.1.2 Methodology

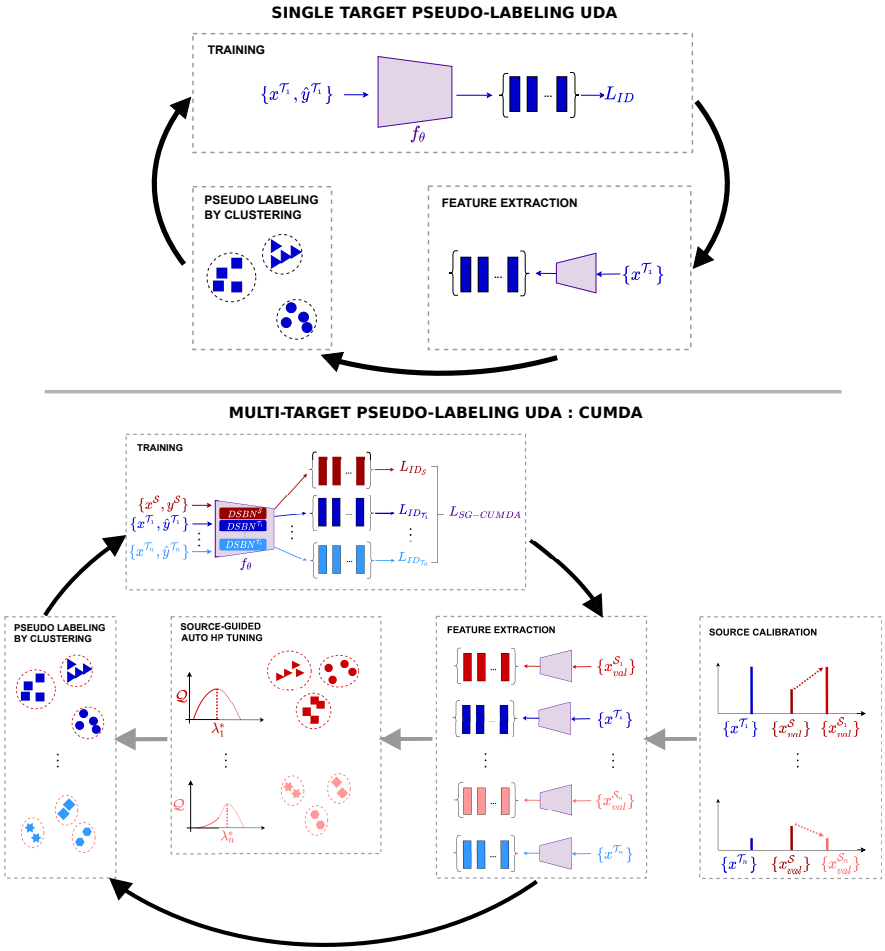


Figure A.1: Upper part: pseudo-labeling paradigm for single-target UDA re-ID. Black arrows indicate the pseudo-labeling and training cycle. *FEATURE EXTRACTION* is carried out for images $\{x^{\mathcal{T}_1}\}$ of the target domain $\mathcal{T}_1$ with a feature encoder $f_\theta$. *PSEUDO LABELING BY CLUSTERING* computes pseudo-labels $\{\hat{y}^{\mathcal{T}_1}\}$ on clustered features. *TRAINING* is done on the pseudo-labeled target set $\{x^{\mathcal{T}_1}, \hat{y}^{\mathcal{T}_1}\}$ by minimizing $\mathrm{L_{ID}}$.
Lower part: proposed pseudo-labeling method for multi-target CUMDA re-ID. Black arrows indicate the pseudo-labeling and training cycle, grey arrows indicate the clustering parameters optimization steps. It considers a set of $n$ target domains $\mathcal{T}_1, ..., \mathcal{T}^n$ and a source domain $\mathscr{S}$. For each target, *SOURCE CALIBRATION* computes an associated labeled source validation set $\{x^{\mathscr{S}_1}_{val}\}$,..., $\{x^{\mathscr{S}_n}\}$. After *FEATURE EXTRACTION* of all source validation and target sets, *SOURCE-GUIDED AUTO HP TUNING* computes target-specific optimal hyperparameter (HP) values $\lambda^*_1, ..., \lambda^*_n$ from calibrated source validation sets my maximizing clustering quality $\mathscr{L}$ (Sec. A.1.2). Target-specific *PSEUDO LABELING BY CLUSTERING* is then carried out. *TRAINING* is done on all pseudo-labeled target sets and on the labeled source domain by minimizing $\mathrm{L_{SG-CUMDA}}$ (Sec. A.1.2).

Our goal is to design a CUMDA re-ID method for cross-domain cattle re-ID.

More specifically, it is expected that this method can:

- specialize for one or multiple target domains to improve performance on them;
- ensure good performance on the source domain.

We also expect such a model to generalize well on an unseen new target domain, as it should accumulate knowledge from multiple domains.

As illustrated in the upper part of Fig. A.1, existing pseudo-labeling methods are designed for UDA re-ID, i.e. to improve re-ID performance on a single target domain, using only data from this domain. Therefore, they need to be rethought in order to meet the previously mentioned objectives of CUMDA re-ID, and to incorporate the use of data from multiple domains motivated by cow re-ID real-world applications. This section introduces key elements of our CUMDA re-ID method. The lower part of Fig. A.1 introduces our CUMDA re-ID method whose components will be motivated hereafter.

**General notations.** We consider a set of $n$ target domains of interest $\mathscr{T}_1, ..., \mathscr{T}_n, n \in \mathbb{N}$ and a source domain $\mathscr{S}$, from which a set of labeled data S from $\mathscr{S}$, and unlabeled data $\mathrm{T}_1, ..., \mathrm{T}_n$ from $\mathscr{T}_1, ..., \mathscr{T}_n$ (the target domains) are available.

**Pseudo-Labeling by Clustering**

A feature encoder $f_\theta, \theta \in \mathbb{R}^p, p \in \mathbb{N}$ (usually a CNN) is trained on the labeled source dataset T, by minimizing a re-ID loss function (e.g.: Classification Loss, Triplet Loss as in [54], a combination of both as in [92]...) $\mathrm{L}_{\mathrm{ID}}(\theta, \mathrm{T})$, w.r.t $\theta$. Then, a clustering function $\mathrm{C}_\lambda$ defined by hyperparameters (HP) $\lambda \in \mathbb{R}^m, m \in \mathbb{N}$ is used to predict pseudo-labels of data samples in each target set, by using the feature representation of their data. Pseudo-labeled target sets $\hat{\mathrm{T}}_1, ..., \hat{\mathrm{T}}_n$ can then be obtained:

$$\forall k \in [1, n], \hat{\mathrm{T}}_k = \mathrm{C}_\lambda(f_\theta, \mathrm{T}_k). \tag{A.1}$$

This step, described by Eq. A.1, is called Pseudo-Labeling by Clustering (PLC). These pseudo-labels will be used to define the loss that supervises the learning on the targets.

**Source-Guided CUMDA re-ID learning**

The objectives of CUMDA re-ID are being able to improve the cross-domain performance for one or multiple target domains, while being able to preserve the source re-ID performance. Inspired by the source-guided loss function designed in Chapter 3 for single-target domain UDA re-ID, we define a new Source-Guided loss function extended for CUMDA re-ID. Therefore, $f_\theta$ is fine-tuned by minimizing a Source-Guided CUMDA (SG-CUMDA) loss function $L_{SG-CUMDA}$ which aggregates all the individual re-ID loss functions on each domain, as follows:

$$L_{SG-CUMDA}(\theta, (S, \hat{T}_1, ..., \hat{T}_n)) = L_{ID}(\theta, S) + \sum_{k=1}^{n} L_{ID}(\theta, \hat{T}_k) \,. \tag{A.2}$$

**Alleviating the domain gap with Domain-Specific Batch Normalization.**

As seen in Chapter 3 and Chapter 4, the gap domain can degrade performance. The proposed methodology proposes to mitigate it at the level of batch normalization layers as done in Chapter 3 with Domain-Specific Batch Normalization (DSBN). As a reminder, DSBN layers has been proposed to be effective for various domain adaptation problems such as UDA classification [12] and UDA re-ID in Chapter 3. It consists in using domain-specific batchnorm affine parameters and computing domain-specific mean and variance. Other network parameters are still shared and used whatever the domain. $f_\theta$ being parameterized by a CNN, DSBN layers are used after each convolutional and fully-connected layers.

**Improving pseudo-labels with Multi-Target Automatic Source-guided selection of Pseudo-Labeling Hyperparameters.**

Pseudo-labeling UDA approaches are sensitive to the quality of the proposed labeling, which depends on the good tuning of clustering hyperparameters $\lambda$. In the context of pedestrian and vehicle re-ID, the ideal $\lambda$ value called $\lambda^*$ has been shown to depend on the target dataset distribution in the feature space, as well as the target dataset statistics (e.g. the number of shots per identity). Most of the works that focus on pedestrian and vehicle UDA re-ID reuse the same values empirically tuned for a specific cross-dataset experiment, on all different cross-datasets considered afterward. It has been shown that this can result in significantly reduced performance compared to choosing a suitable value.

For real world cow-re-ID, because the target is unlabeled, it is impossible to build a labeled validation set to tune this value. Besides, usual $\lambda$ value used for person re-ID may not translate well to cow re-ID problem, given the particularities of cow datasets (color distribution, viewpoints, ...). Therefore, we propose to automate the tuning of $\lambda$ from the labeled data. To do so, we use the HyPASS paradigm. HyPASS estimates by model selection on $C_\lambda$, the value $\lambda^*$ such as $\lambda^* = \arg\max_\lambda \mathcal{Q}(C_\lambda, S^{val})$ where $\mathcal{Q}$ is a clustering quality function and $S^{val}$ a labeled validation set from the source data. It is illustrated on Fig. A.1 as source-guided auto HP tuning.

**HyPASS-SC for CUMDA: improving the robustness to domain gap.** In this section, we adapt HyPASS to our cow re-ID CUMDA problem, by selecting a specific value $\lambda_k$ for each target dataset $T_k$. The PLC defined by Eq. A.1, is redefined as a Domain-specific PLC given by:

$$\forall k \in [1, n], \hat{T}_k = C_{\lambda_k}(f_\theta, T_k). \tag{A.3}$$

HyPASS functioning relies on domain-gap reduction. In the multi-target use-case, we propose to achieve it in two ways:

- At the feature level, target-specific DSBN is leveraged to reduce the domain-gap in the feature space (cf. section A.1.2);
- At the dataset statistics level, a new Source validation Calibration (SC) approach is proposed.

While cross-dataset statistic gap may be overlooked for an academic person and vehicle re-ID such as HyPASS as in the previous sections, it becomes crucial in the considered cow re-ID problematic. Indeed, in the well-known person re-ID datasets, usual statistics discrepancies are minimal, with between 21 and 36 shots per ID (e.g. Market1501 [172], DukeMTMC [111], personX [123] and MSMT17 [140]). However that does not hold true in all cases. Especially in animal-related re-ID, where the data may be difficult to acquire resulting in higher discrepancies in shots per ID. Moreover, contrary to person or vehicle re-ID applications in open-world, the number of cows in a farm of interest is generally known, or can be easily estimated for a cross-domain re-ID applications. This allows us to design SC for

| Dataset | # train IDs | # train images | # test IDs | # query images | # gallery images |
|---|---|---|---|---|---|
| Cows2021 * | 90 | 4602 | 91 | 855 | 3213 |
| HolsteinCattle * | 68 | 609 | 68 | 204 | 414 |
| CowFisheye | 62 | 6334 | 16 | 151 | 2224 |

Table A.1: Dataset statistics. For Cows2021 [40] and HolsteinCattle [10], * indicates that the dataset is extracted from the RGB annotated portion of the complete dataset, following a 50/50 ID split for Train/Test. There is no overlap between train IDs and test IDs (cf. Sec. A.1.3).


Figure A.2: Illustration of the content of each dataset. From left to right: Cows2021 [40], HolsteinCattleRecognition [10], CowFisheye (private).

.

cattle re-ID, which consists in equalizing the number of shots per ID of the source to match that of the target. SC generates target-specific source validation sets $S_k^{val}$ from $S^{val}$, that reduce the cross-dataset statistics gap with the corresponding target training set $T_k$. SC is represented as Source Calibration on Fig. A.1. HyPASS is then run on $S_k^{val}$ to compute $\lambda_k^*$, the optimal hyperparameter value for clustering on $T_k$. The combined use of HyPASS and SC will be referred to as HyPASS-SC in the rest of the section. Implementation details are given in Sec. A.1.3.

### A.1.3 Experiments

#### Datasets

In this section, we employ three different datasets: Cows2021 [40], HolsteinCattleRecognition [10] and the private dataset CowFisheye. The content of each dataset is illustrated in Fig. A.2.

**Cows2021**

Cows2021 [40] is a dataset featuring RGB images and videos of 186 individuals. The data was acquired from 4 m above the ground by a pinhole camera pointed downwards.

**Image extraction.** The images were extracted over one month of acquisition. The extraction of cow images from the video stream relies on an oriented bounding-box detector and a tracker. The boxes are centered around cow torsos, excluding their heads, with all individuals facing right. For more details on data acquisition, please refer to [40].

**IDs & samples.** In this study, labeled annotations for initial supervised training are needed for relevant performance comparisons with unsupervised UDA. Therefore, only its labeled data is used. A total of 8670 images depicting 181 distinct individuals were extracted, for an average of 48 shots per identity. More details on image repartition can be found in Tab. A.1.

**Complexity.** Despite an acquisition that spreads over one month, the illumination and viewpoint of the cows vary little between acquisitions. There is little to no occlusion in the images.

**HolsteinCattleRecognition**

HolsteinCattleRecognition [10] is a dataset featuring RGB and infrared images of 1237 individuals. The data was acquired by a pinhole camera placed at gound level, 5 m away from the milking machine it films. For concision, we refer to it as HolsteinCattle in the rest of the section.

**Image extraction.** The images were extracted over nine days of acquisition. Each of them contains a single cow in the milking machine. For more details on data acquisition, please refer to [10].

**IDs & samples.** In this study, only the RGB data is used. A total of 1227 images depicting 136 distinct individuals were extracted, for an average of 9 shots per identity. More details on image repartition can be found in Tab. A.1.

**Complexity.** This dataset features partially occluded cattle positioned differently in the milking machine.

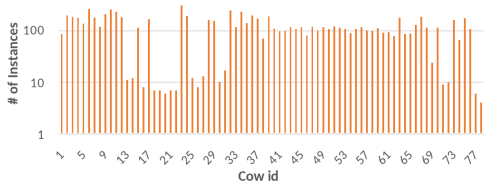


Figure A.3: Number of images per ID in CowFisheye dataset.

**CowFisheye**

CowFisheye is a private dataset featuring RGB images of 78 individuals. It was acquired from a single farm from 4 fisheye cameras pointing downwards, positioned 6 m above the ground. The acquisition was done during both day and night at 25 fps at a resolution of $4000 \times 3000$ pixels. Identities were annoted manually. This dataset reflects the usual challenging data encountered in the use case where cows must be identified 24/7 wherever they are in the farm.

**Image extraction.** Images were extracted over 6 days of acquisition, during both day and night. The extraction was done automatically by a detector, during periods when there were cow movements. The resulting images contain the whole cow body and head, horizontally aligned and facing right.

**IDs & samples.** A manual selection and annotation of images was done to ensure a good variability of viewpoints and lighting conditions for each cow. A total of 8709 images depicting 78 distinct individuals were extracted for an average of 112 instances per identity. More details on images repartition can be found in Tab. A.1 and Fig. A.3.

**Complexity.** The CowFisheye dataset complexity reflects the desired application: re-ID for 24/7 monitoring of cows in the whole camera network. More precisely, the camera configuration that required for coverage of the whole farm, may introduce significant cow occlusion by obstacles or other individuals. Distortions inherent to fisheye cameras are also present. Besides, the acquisition is done with varied illuminations, which can cause significant discrepancies in the appearance of a cow. At night specifically, a near-infrared mode is activated resulting in black and white pictures. An illustration of the complexity of the CowFisheye dataset is proposed in Fig. A.4.

Figure A.4: Illustration of the complexity of the CowFisheye dataset. Left: varied lighting conditions, Center: occlusion by objects/cows, Right: varied viewpoints. All pictures represent the same individual.

### Experimental settings

#### Use case

Our use case is specific to re-ID of animals in multiple farms, with a labeled, and one or many unlabeled farms. The objective of our CUMDA re-ID method presented in Sec. A.1.2 is twofold:

- conservation on the source (labeled) domain;
- specialization on the target (unlabeled) domains.

We also expect better generalization on a new unseen target domain which would correspond to a new farm in a real world application.

Therefore, for each training, performance on all three datasets is reported. Throughout the experiments, the term "good practices" refers to the application of source guidance (cf. Sec. A.1.2), DSBN (cf. Sec. A.1.2) and HyPASS-SC (cf. Sec. A.1.2). Please note that this work does not aim at optimizing the network architecture or learning parameters, it rather establishes good practices for efficient CUMDA re-ID by pseudo-labeling.

#### Framework

In order to introduce some robustness to the pseudo-labels noise, we use the state-of-the-art framework Mutual Mean Teaching (MMT) [41] paired with a Resnet-18 [52] backbone pretrained on ImageNet [26]. The last stride of the Resnet-18 is set to 1 to increase the feature map resolution. The DBSCAN clustering algorithm is run on the $k$-reciprocal encoded features with $k = 30$. DBSCAN parameters $n_{min}$ and $\epsilon$ are set to $n_{min} = 0.4$ and $\epsilon = 0.6$, the usual optimal value [31]. Their values remain constant, except when $\epsilon$ is optimized by HyPASS-SC. All other unspecified values are

set similarly to the original MMT paper [41]. The work conducted in this section is however not limited to MMT and could be extended to any UDA framework.

### Data preprocessing

For each domain, mini-batches of size 16 are built with P=4 identities and K=4 shots per identity. CUMDA batches may vary in size as they are constructed from one mini-batch for the source dataset and one for each target dataset, when applicable. Thus, their size depends on $n_{domain}$, the number of domains used during training. Here, the batch size is equal to $16 \times n_{domain}$. Images are resized to 128 × 128 pixels. Re-ID related data augmentations such as crop and flip are applied during the training stage. Random erasing was not applied because it experimentally decreased the cow re-ID performance.

### Initial training

The network is trained during 20 epochs of 200 iterations each on the chosen source dataset. The learning rate is set to $lr = 3.5.10^{-4}$. Both triplet and cross-entropy losses are used during the training on source images [41].

Table A.2: Performance (in %) of models supervised on a single source dataset and direct transfer on each target dataset without adaptation.

| Method | Train | | Test | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Source | Target | Cows2021 | | HolsteinCattle | | CowFisheye | |
| | | | mAP | rank-1 | mAP | rank-1 | mAP | rank-1 |
| Supervised training | Cows2021 | None | **95.3** | **98.2** | 8.0 | 13.7 | 16.8 | 45.7 |
| Supervised training | HolsteinCattle | None | 29.1 | 73.1 | **81.2** | **91.7** | 12.7 | 25.2 |
| Supervised training | CowFisheye | None | 71.0 | 95.1 | 13.6 | 24.5 | **50.5** | **75.5** |

### Domain adaptation

The network is trained during 15 epochs of 200 iterations. This choice is driven to avoid overfitting on the smaller datasets. The learning rate is set to $lr = 3.5.10^{-4}$. Both triplet and cross-entropy losses are used during the training. Source and target share the same fully connected layer for classification. When using MMT, testing is systematically done on model n°1 as in real-world applications, determining which

model performs best on the target is impossible. Indeed, the target dataset is not annotated. Concerning the adaptation on multiple target datasets, when applicable, DSBN is generalized so as to have one BN per domain. During testing, the BN of the domain that is most similar in appearance to the tested domain is used. More specifically, the test domain BN is used if it has been computed during training. Otherwise, the BN of CowFisheye is used when testing on Cows2021 or HolsteinCattle, and the BN of Cows2021 is used when testing on CowFisheye.

**Testing**

Because most datasets are extracted from a unique camera, the evaluation is done without filtering images from the same camera. The mean Average Precision (mAP) is reported as evaluation metric. It is an indicator of the network ability to correctly identify a query individual, among individuals in a gallery, and should be maximized. No re-ranking is applied during testing. The ID splitting for Cows2021 and HolsteinCattle, is done following the original ascending numbering. The first half of the identities is taken as train set and the other half as test set.

**Performance computation detailed.**

In this section, Tab. A.5 - A.8 exhibit the variation in performance of the network, for each ablation step. The $\Delta_{m\text{AP}}$ is computed under the following protocol. Let us consider a set of n domains $\mathscr{D}_1$, ..., $\mathscr{D}_n$, and let us test on the domain $\mathscr{D}_i$, $i \in \{1,...,n\}$.

If $\mathscr{D}_i$ is tested as a source, the performance for all cross-domain experiments $\mathscr{D}_i \rightarrow \mathscr{D}_k$, $k \in \{1,...,n\} \backslash \{i\}$, is compared to the network supervised on $\mathscr{D}_i$. The result is then averaged.

If $\mathscr{D}_i$ is tested as a target, the performance for all cross-domain experiments $\mathscr{D}_k \rightarrow \mathscr{D}_i$, $k \in \{1,...,n\} \backslash \{i\}$, is compared to the network supervised on $\mathscr{D}_k$. The result is then averaged.

If $\mathscr{D}_i$ is tested as an unseen dataset, the performance for all cross-domain experiments $\mathscr{D}_k \rightarrow \mathscr{D}_m$, $k, m \in \{1,...,n\} \backslash \{i\}$ with $k \neq m$, is compared to the networked supervised on $\mathscr{D}_k$. The result is then averaged.

For the sake of clarity, let us detail the computation of the first line and first column of Tab. A.5: testing the baseline UDA (MMT) on Cows2021 as a source dataset. The baseline mAP on Cows2021 as a source is equal to 84.1% and 39.2%, for the cross domains Cows2021→CowFisheye and Cows2021→HolsteinCattle respectively. A

supervised network on Cows2021 has a mAP of 95.3% on Cows2021 (cf. Tab. A.2). Therefore, the $\Delta_{m\text{AP}}$ is equal to $((84.1-95.3)+(39.2-95.3))/2 = -33.7$ p.p..

**HyPASS-SC**

HyPASS-SC optimizes the value of DBSCAN hyperparameter $\epsilon$ in the range $[0.35, 0.65]$, which is the range of acceptable values for human datasets applications [31]. There is an order of magnitude difference in number of shots per individual between HolsteinCattle (9) and other datasets (48 and 112). Therefore, when dealing with HolsteinCattle, HyPASS-SC computes a subsampled or oversampled source validation dataset so that the number of shots per identities in the source validation set roughly matches the one of the target, as described in Sec. A.1.2. Impact on the performance of shots leveling will be shown in Sec. A.1.6. Random subsampling Cows2021 or CowFisheye is straightforward, while oversampling HolsteinCattle is done by applying the same data augmentation than for training. In this work, subsampling aims to attain 9 shots/ID regardless of the source, while oversampling is done to reach 90 shots per identity, regardless of the target. Please note that the source data used for training is not impacted by this step. In a real-world application, the number of cows in a farm is known and the number of instances per identity can be approximated by dividing the number of acquired images by the estimated number of cows in the exploitation.

### A.1.4 Results

**Effectiveness of our CUMDA with single target domain**

**Supervised training & direct transfer.** Supervised training results can be found in Tab. A.2. These results, when training and testing on the same dataset, give an idea of the complexity of each dataset. from highest to lowest: CowFisheye, HolsteinCattle and Cows2021.

We also show the cross-domain performance of models supervised on a source dataset and directly evaluated on the other datasets, without adaptation. The low performance on these datasets demonstrates the need for domain adaptation. In Tab. A.2, two applications stand out. First, the direct transfer Cows2021→HolsteinCattle shows the poorest performance at 8.0% which indicates a strong domain gap. Second, the direct transfer CowFisheye→Cows2021 shows the highest performance at 71.0%, which indicates a smaller domain gap.

Table A.3: Domain adaptation baseline. mAP (in %), Δ (in p.p.) indicates the difference with initial supervised models (cf Tab. A.2).

| Method | Train | | Test | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Source | Target | Cows2021 | | HolsteinCattle | | CowFisheye | |
| | | | mAP | $\Delta_{mAP}$ | mAP | $\Delta_{mAP}$ | mAP | $\Delta_{mAP}$ |
| baseline | Cows2021 | HolsteinCattle | 39.2 | -56.1 | 12.6 | +4.6 | 9.1 | -7.7 |
| baseline | Cows2021 | CowFisheye | 84.1 | -11.2 | 10.8 | +2.8 | 24.3 | +7.5 |
| baseline | HolsteinCattle | Cows2021 | 84.6 | +55.5 | 25.5 | -55.7 | 16.0 | +3.3 |
| baseline | HolsteinCattle | CowFisheye | 54.9 | +25.8 | 19.6 | -61.6 | 14.4 | +1.7 |
| baseline | CowFisheye | Cows2021 | 88.9 | +17.9 | 8.5 | -5.1 | 19.9 | -30.6 |
| baseline | CowFisheye | HolsteinCattle | 30.8 | -40.2 | 22.7 | +9.1 | 14.3 | -36.2 |

Table A.4: Domain adaptation with good practices (source guidance, DSBN and HyPASS-SC). mAP (in %), Δ (in p.p.) indicates the difference with initial supervised models (cf Tab. A.2). **g.p.**: good practices.

| Method | Train | | Test | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Source | Target | Cows2021 | | HolsteinCattle | | CowFisheye | |
| | | | mAP | $\Delta_{mAP}$ | mAP | $\Delta_{mAP}$ | mAP | $\Delta_{mAP}$ |
| g.p. | Cows2021 | HolsteinCattle | 95.4 | +0.1 | 15.9 | +7.9 | 20.1 | +3.3 |
| g.p. | Cows2021 | CowFisheye | 95.0 | -0.3 | 13.7 | +5.7 | 28.2 | +11.4 |
| g.p. | HolsteinCattle | Cows2021 | 87.2 | +58.1 | 80.8 | -0.4 | 15.3 | +2.6 |
| g.p. | HolsteinCattle | CowFisheye | 57.9 | +28.8 | 81.1 | -0.1 | 16.8 | +4.1 |
| g.p. | CowFisheye | Cows2021 | 90.6 | +19.6 | 13.3 | -0.3 | 60.1 | +9.6 |
| g.p. | CowFisheye | HolsteinCattle | 70.9 | -0.1 | 38.6 | +25.0 | 60.1 | +9.6 |

**UDA baseline.** In the rest of the section, we will refer to domain adaptation without source guidance as “baseline". Its functioning is illustrated in the upper part of Fig. A.1. As shown in Tab. A.3, the performance on the target dataset increases with $\Delta_{mAP}$ values in [+1.7 p.p., +55.5 p.p.]. This shows that the domain adaptation is efficient on the target dataset for all presented cases. However, the decrease seen in all diagonal elements of Tab. A.3, indicate that the source dataset is partially forgotten by the network. This effect is drastic with $\Delta_{mAP}$ values in [-61.6 p.p., -11.2 p.p.].

The generalization performance on an unseen dataset is inconsistent and seems to evolve towards that of a supervised network that is supervised on the target dataset. For example, in the case CowFisheye→HolsteinCattle, the performance of

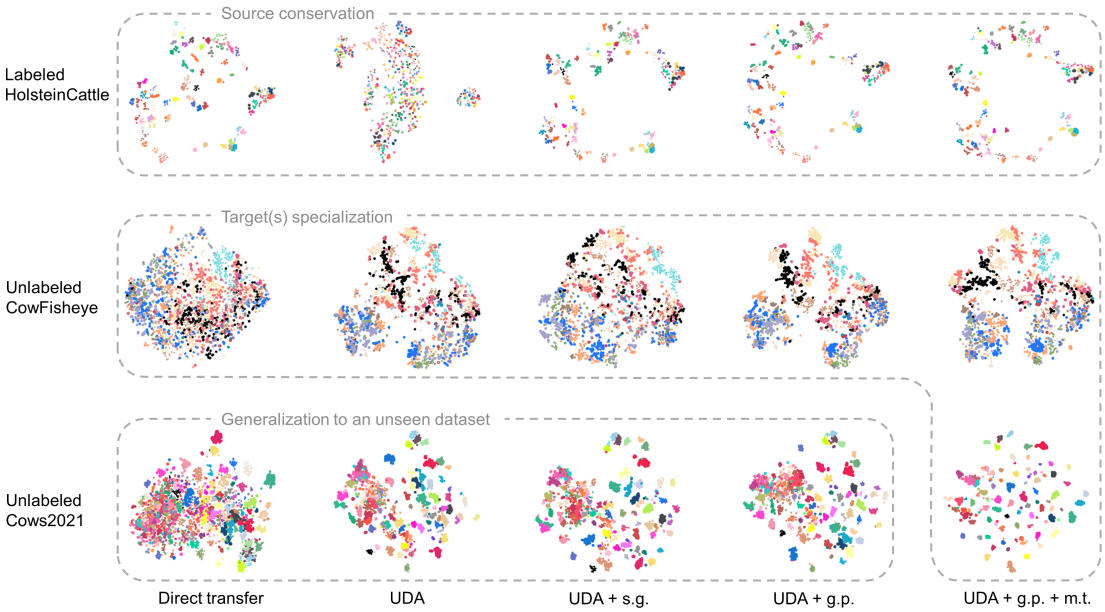


Figure A.5: Evolution of the embedding space on all validation datasets for the cross-domain HolsteinCattle→CowFisheye. Visualization with t-SNE where each identity is assigned a random color and size. Each point represents an image in the embedding space. Each row correspond to a dataset. Each column corresponds to a use-case. Acronyms. **UDA**: UDA baseline w/ MMT, **s.g.**: source guided, **g.p.**: good practices, **m.t.**: multi-target. Ideally, points of the same colors should be clustered and well separated from all other clusters. Best viewed in color.

the network on Cows2021 decreases from 71.0% (cf. Tab. A.2) before UDA to 30.8% after (cf. Tab. A.3). This performance ressembles the 29.1% performance seen for a network solely supervised on HolsteinCattle (cf. Tab. A.2). In conclusion, the baseline approach adapts the network to a single dataset, without ensuring performance gains on any other dataset.

**CUMDA good practices.** In the rest of the section, we will refer to domain adaptation with source guidance, DSBN and HyPASS-SC as “good practices". Its functioning is illustrated in the lower part of Fig. A.1. The results with good practices are reported in Tab. A.4. The improvements over the baseline (cf. Tab. A.3) are multiple.

First, it outperforms the supervised network on source, target and a third domain in a consistent way. On the source domain, the performance is equivalent or better, with a +9.6 p.p. increase in terms of mAP for CowFisheye. On the target domain, the performance increases is drastic with an average $\Delta_{mAP}$ of $(58.1+19.6)/2 =$ +38.9 p.p., +16.5 p.p. and +7.8 p.p. on Cows2021, HolsteinCattle and CowFisheye when they are taken as target domains respectively. Generalization performance on

Table A.5: Relative performance of the baseline, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | -33.7 | -58.7 | -33.4 |
| Target | +36.7 | +6.9 | +4.6 |
| Unseen | -7.2 | -1.2 | -2.2 |

the unseen domain increases on average of $(28.8-0.1)/2 = +14.4$ p.p., +2.7 p.p. and +3.0 p.p. on the same datasets.

Second, on the target domain, our proposed good practices outperform the baseline. To characterise this, we compute the increment in performance between Tables A.4 and A.3. On the source domain, the performance increase is drastic with values as high as $-0.1+61.6 = +61.5$ p.p. for the cross domain HolsteinCattle→CowFisheye. On the target domain, the performance increases consistently with values in [+1.7 p.p., +15.9p.p.]. Generalization performance on the unseen domain increases with values in [-0.7 p.p., +40.1 p.p.].

All these results demonstrate the network ability to both remember the source dataset and leverage information from all domains to increase performance steadily on all domains. Therefore, we recommend the proposed good practices, for conservation on the source domain, better specialization on each seen domain and better generalization on unseen domains.

### A.1.5 Ablation study with single target

In section A.1.4, we have demonstrated the performance gains brought by our proposed good practices for re-ID CUMDA over direct transfer and baseline UDA domain adaptation. In this section, the relative importance of all deployed good practices is investigated through an ablation study. Averaged performance variations with respect to a network supervised on source are reported in Tab. A.5 - A.8. The good practices in this section refers to the use of source-guidance, DSBN and HyPASS-SC.

**Source guidance.** Averaged performance with source guidance is reported in

Table A.6: Relative performance of source-guided UDA, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | -0.4 | -0.6 | +9.2 |
| Target | +36.6 | +6.0 | +5.7 |
| Unseen | +14.2 | +1.1 | +3.3 |

Tab. A.6. We compare these results to those of the baseline, reported in Tab. A.5.

Providing the source as labeled data during training increases the performance drastically on the source dataset. Compared to the baseline, the average performance increase is equal to $33.7 - 0.4 = +33.3$ p.p., +58.1 p.p. and +42.6 p.p. when considering Cows2021, HolsteinCattle and CowFisheye as source respectively. However, the performance on target dataset is unchanged with an average delta in performance of $-0.1$ p.p., $-0.9$ p.p. and +1.1 p.p.. The model better generalizes thanks to the knowledge of both source and target domains with an increase of +21.4 p.p., +2.3 p.p. and +5.5 p.p. on Cows2021, HolsteinCattle and CowFisheye respectively.

In summary, compared to the baseline, the source guidance allows the model to perform similarly on the target while ensuring good conservation of the source. Benefiting from the information of both source and target, the model better generalizes to an unseen dataset.

**DSBN.** In this section, alleviating the domain gap is achieved with the use of DSBN. Averaged performance with source guidance + DSBN is reported in Tab. A.7. We compare these results to those of the source guided approach, reported in Tab. A.6.

When compared to source-guided UDA, there is significant performance increase on the target of $40.2 - 36.6 = +3.6$ p.p., +4.0 p.p. and +2.5 p.p. for Cows2021, HolsteinCattle and CowFisheye respectively. However, performance on the source dataset slightly decreases with deltas of +0.7 p.p., $-1.8$ p.p. and $-1.7$ p.p.. Overall, the generalization to an unseen dataset is unchanged as the effects of DSBN cancel out.

In summary, in our experiments, DSBN does not seem to guarantee better cow re-identification. However, we will see that it is useful by allowing the use of HyPASS.

Table A.7: Relative performance of source-guided + DSBN UDA, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | +0.3 | -2.4 | +7.5 |
| Target | +40.2 | +10.0 | +8.2 |
| Unseen | +13.0 | +3.1 | +2.5 |

**HyPASS-SC.**

The use of DSBN allows for source-guided selection of pseudo-labeling hyperparameters, achieved here with HyPASS-SC. Averaged performance with source guidance + DSBN + HyPASS-SC is reported in Tab. A.8. We compare these results to those of the source-guided + DSBN approach, reported in Tab. A.7.

In comparison with source-guided + DSBN UDA, the performance on the target domain HolsteinCattle increases of $16.5 - 10.0 = +6.5$ p.p.. This may be explained by the significant differences between HolsteinCattle and other domains, which may result in a significant shift of the optimal value of clustering parameters. On the Cows2021 dataset, HyPASS-SC seems to performs slightly worse than source-guided + DSBN with a difference of $-1.3$ p.p.. This seems to indicate that HyPASS-SC may not be optimal in all cases, especially when the source test set has few images. However, HyPASS-SC retains its usage by removing the need for user-set parameters. On the source domain, the performance increases with deltas of $-0.1 - 0.3 = -0.4$ p.p., +2.1 p.p. and +2.1 p.p.. It even exceeds the performance of source-guided UDA (cf. Tab. A.6). The generalization performance is increased of +1.4 p.p., $-0.4$ p.p. and +0.5 p.p..

In conclusion, we find that HyPASS-SC has a positive effect on performance on all datasets. Indeed, it ensures good clustering on all targets. This allows for better source conservation, target specialization and generalization on an unseen dataset than the other approaches presented here.

For the cross-domain HolsteinCattle→CowFisheye, a t-SNE visualization of the effects of domain adaptation, source-guidance and good practices on the embedding space, is proposed in Fig. A.5. It shows: the decreased performance on source with the baseline, the increased performance on all datasets with source guidance

Table A.8: Relative performance of our good practices single target CUMDA, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | -0.1 | -0.3 | +9.6 |
| Target | +38.9 | +16.5 | +7.8 |
| Unseen | +14.4 | +2.7 | +3.0 |

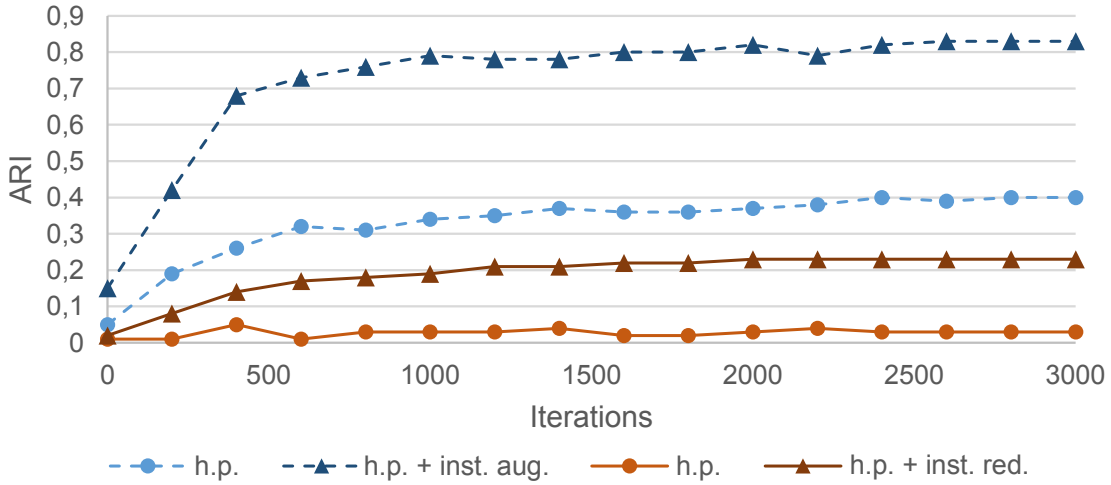


Figure A.6: Evolution of the Adjusted Random Index (ARI) of target clustering. Influence of the calibration of source validation set on HyPASS performances. In blue and dashed lines HolsteinCattle→Cows2021, in red, CowFisheye→HolsteinCattle. Acronyms. **h.p.**: HyPASS [31], **inst. red.**: source instance reduction, **inst. aug.**: source instance augmentation.

and the best performances obtained with the proposed good practices.

CUMDA re-ID with multiple-targets (CowFisheye and Cows2021) is also illustrated with clear gains on all three datasets. More evaluations are presented in Sec. A.1.7.

### A.1.6 Benefit of the source calibration in HyPASS-SC

As explained in section A.1.2, HyPASS is sensitive to dataset statistics. More specifically, in our case, to the number of shots per identity of Cows2021 (48), HolsteinCattle(9) and CowFisheye (112).

To validate the importance of our proposed source validation set calibration, we compare the performance of HyPASS-SC and HyPASS on the cross domains HolsteinCattle→Cows2021 (oversampling use-case) and CowFisheye→HolsteinCattle (subsampling use-case).

The quality of the clustering is evaluated with the Adjusted Random Index (ARI) which is a measure of the similarity between two data clusterings. It is computed between the target training set labels and the cluster predictions, using the scikit-learn implementation[1]. Fig. A.6 illustrates the evolution of ARI when leveling the source and target statistics. Higher values of ARI indicate a better clustering. Performance is compared at the $3000^{th}$iteration.

In the cross-domain HolsteinCattle→Cows2021, an oversampling of HolsteinCattle from 9 shots/ID to around 90 shots/ID is carried out. As a result, the ARI doubles, increasing from 0.40 without calibration, to 0.83 with it. In terms of mAP, the performance on the target Cows2021 increases from 77.4 % without calibration, to 87.2 % with it.

In the cross-domain CowFisheye→HolsteinCattle, a subsampling of CowFisheye from 112 shots/ID to around 9 shots/ID is carried out. The resulting ARI increase is substantial, evolving from 0.03 without calibration, to 0.23 with it. In terms of mAP, the performance on the target HolsteinCattle increases from 29.3 % without calibration, to 38.6 % with it.

These results show the importance of source validation set calibration in the case of datasets with highly different number of shots per ID, which can be a recurrent issue when dealing with animal datasets. For information, the calibration of the source has been systematically applied on all aforementioned experiments.

### A.1.7 Effectiveness of our CUMDA with multiple targets

One of our goals is to generalize the domain adaptation to multiple target domains. This use-case reflects real-world needs where, from a labeled dataset, the model should be adapted to multiple farming exploitations. Averaged performance or our CUMDA approach including good practices (source guidance, DSBN and HyPASS-SC) is reported in Tab. A.10. We compare these results to those of the source-guided approach, reported in Tab. A.9.

[1]https://scikit-learn.org/

Table A.9: Relative performance of source-guided + multi-target UDA, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | -0.3 | -1.5 | +10.5 |
| Target | +35.0 | +7.4 | +3.8 |

Table A.10: Relative performance of good practices + multi-target CUMDA, compared to direct transfer. $\Delta_{mAP}$ (in p.p.) indicates the difference of mAP (in %) with direct transfer (cf. Tab. A.2)

| | Test | | |
|---|---|---|---|
| | Cows2021 | HolsteinCattle | CowFisheye |
| Test set | $\Delta_{mAP}$ | $\Delta_{mAP}$ | $\Delta_{mAP}$ |
| Source | 0.0 | -0.9 | +10.4 |
| Target | +38.8 | +13.4 | +7.5 |

On the target datasets, our approach outperforms the source-guided UDA approach with $\Delta_{mAP}$ of $38.8 - 35.0 = +3.8$ p.p., +6.0 p.p. and +3.7 p.p. for Cows2021, HolsteinCattle and CowFisheye respectively. Besides, the performance on the source dataset is conserved. This demonstrate the importance of the proposed good practices when it comes to multiple datasets application. This performance increase can be explained by the complementarity of DSBN and HyPASS-SC.

DSBN allows some domain gap alleviation through domain specific normalization. It also authorizes domains to share the same backbone, which helps generalization. HyPASS-SC provides optimized clustering parameters on each target dataset, depending on its statistics. This ensures good clustering quality on the target domains, which in turn increases the network performance on all datasets.

# Bibliography


[1] W. Andrew, J. Gao, S. Mullan, N. Campbell, A. W. Dowsey, and T. Burghardt. Visual identification of individual holstein-friesian cattle via deep metric learning. *Computers and Electronics in Agriculture*, 185:106133, 2021.

[2] W. Andrew, C. Greatwood, and T. Burghardt. Visual localisation and individual identification of holstein friesian cattle via deep learning. In *IEEE International Conference on Computer Vision (ICCV) Workshops*, pages 2850–2859, 2017.

[3] W. Andrew, S. Hannuna, N. Campbell, and T. Burghardt. Automatic individual holstein friesian cattle identification via selective local coat pattern matching in rgb-d imagery. In *IEEE International Conference on Image Processing (ICIP)*, pages 484–488. IEEE, 2016.

[4] M. Anthony, P. L. Bartlett, and P. L. Bartlett. *Neural network learning: Theoretical foundations*, volume 9. cambridge university press Cambridge, 1999.

[5] T. G. authors. Gpyopt: A bayesian optimization framework in python. `http://github.com/SheffieldML/GPyOpt`, 2016.

[6] S. Bak, P. Carr, and J.-F. Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. In *ECCV*, pages 189–205, 2018.

[7] D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2000.

[8] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[9] L. Bergamini, A. Porrello, A. C. Dondona, E. Del Negro, M. Mattioli, N. D'alterio, and S. Calderara. Multi-views embedding for cattle re-identification. In *IEEE International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 184–191, 2018.

[10] A. Bhole, O. Falzon, M. Biehl, and G. Azzopardi. A computer vision pipeline that uses thermal and rgb images for the recognition of holstein cattle. In *International Conference on Computer Analysis of Images and Patterns*, pages 108–119. Springer, 2019.

[11] C. Bliss, W. Cochran, and J. Tukey. A rejection criterion based upon the range. *Biometrika*, 43(3/4):418–422, 1956.

[12] W.-G. Chang, T. You, S. Seo, S. Kwak, and B. Han. Domain-specific batch normalization for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7354–7362, 2019.

[13] X. Chang, Y. Yang, T. Xiang, and T. M. Hospedales. Disjoint label space transfer learning with common factorised space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[14] G. Chen, Y. Lu, J. Lu, and J. Zhou. Deep credible metric learning for unsupervised domain adaptation person re-identification. In *European Conference on Computer Vision (ECCV)*, 2020.

[15] H. Chen, B. Lagadec, and F. Bremond. Ice: Inter-instance contrastive encoding for unsupervised person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14960–14969, 2021.

[16] H. Chen, Y. Wang, B. Lagadec, A. Dantcheva, and F. Bremond. Joint generative and contrastive learning for unsupervised person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[17] Y. Chen, X. Zhu, and S. Gong. Instance-guided context rendering for cross-domain person re-identification. In *ICCV*, pages 232–242, 2019.

[18] Z. Chen, J. Zhuang, X. Liang, and L. Lin. Blending-target domain adaptation by adversarial meta-adaptation networks. In *Proceedings of the IEEE/CVF*

*Conference on Computer Vision and Pattern Recognition (ICCV)*, pages 2248–2257, 2019.

[19] S. Choi, T. Kim, M. Jeong, H. Park, and C. Kim. Meta batch-instance normalization for generalizable person re-identification. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 3425–3435, 2021.

[20] C. Cortes, Y. Mansour, and M. Mohri. Learning bounds for importance weighting. In *Nips*, 2010.

[21] M. Crucianu, M. Ferecatu, and N. Boujemaa. Relevance feedback for image retrieval: a short survey. *Report of the DELOS2 European Network of Excellence (FP6)*, 2004.

[22] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 2013.

[23] Y. Dai, X. Li, J. Liu, Z. Tong, and L.-Y. Duan. Generalizable person re-identification with relevance-aware mixture of experts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16145–16154, 2021.

[24] Y. Dai, J. Liu, Y. Bai, Z. Tong, and L.-Y. Duan. Dual-refinement: Joint label and feature refinement for unsupervised domain adaptive person re-identification. *IEEE Transactions on Image Processing*, 2021.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 2009.

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[27] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[28] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *CVPR*, pages 994–1003, 2018.

[29] F. Dubourvieux, R. Audigier, A. Loesch, S. Ainouz, and S. Canu. A formal approach to good practices in pseudo-labeling for unsupervised domain adaptive re-identification. *arXiv preprint arXiv:2112.12887*, 2021.

[30] F. Dubourvieux, R. Audigier, A. Loesch, S. Ainouz, and S. Canu. Unsupervised domain adaptation for person re-identification through source-guided pseudo-labeling. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021.

[31] F. Dubourvieux, A. Loesch, R. Audigier, S. Ainouz, and S. Canu. Improving unsupervised domain adaptive re-identification via source-guided selection of pseudo-labeling hyperparameters. *IEEE Access*, 9:149780–149795, 2021.

[32] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, 1996.

[33] H. Fan, L. Zheng, and Y. Yang. Unsupervised person re-identification: Clustering and fine-tuning. *arXiv preprint arXiv:1705.10444*, 2017.

[34] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, pages 2360–2367, 2010.

[35] H. Feng, M. Chen, J. Hu, D. Shen, H. Liu, and D. Cai. Complementary pseudo labels for unsupervised domain adaptation on person re-identification. *IEEE Transactions on Image Processing*, 2021.

[36] M. Ferecatu, N. Boujemaa, and M. Crucianu. Semantic interactive image retrieval combining visual and conceptual content description. *Multimedia systems*, 13(5):309–322, 2008.

[37] Y. Fu, Y. Wei, G. Wang, Y. Zhou, H. Shi, and T. S. Huang. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *ICCV*, pages 6112–6121, 2019.

[38] Y. Fu, Y. Wei, G. Wang, Y. Zhou, H. Shi, and T. S. Huang. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[39] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

[40] J. Gao, T. Burghardt, W. Andrew, A. W. Dowsey, and N. W. Campbell. Towards self-supervision for video identification of individual holstein-friesian cattle: The Cows2021 dataset. *arXiv preprint arXiv:2105.01938*, 2021.

[41] Y. Ge, D. Chen, and H. Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In *International Conference on Learning Representations*, 2019.

[42] Y. Ge, F. Zhu, D. Chen, R. Zhao, and h. Li. Self-paced contrastive learning with hybrid memory for domain adaptive object re-id. In *Advances in Neural Information Processing Systems*, 2020.

[43] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic. Unsupervised multi-target domain adaptation: An information theoretic approach. *IEEE Transactions on Image Processing (TIP)*, 29:3993–4002, 2020.

[44] A. Ghosh, H. Kumar, and P. Sastry. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[45] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

[46] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*, pages 262–275, 2008.

[47] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[48] F. M. Hafner, A. Bhuiyan, J. F. Kooij, and E. Granger. Rgb-depth cross-modal person re-identification. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–8. IEEE, 2019.

[49] O. Hamdoun, F. Moutarde, B. Stanciulescu, and B. Steux. Person re-identification in multi-camera system by signature based on interest point descriptors collected on short video sequences. In *2008 Second ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–6, 2008.

[50] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[51] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[52] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

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

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

[55] M. Hirzer, P. M. Roth, M. Köstinger, and H. Bischof. Relaxed pairwise learned metric for person re-identification. In *ECCV*, pages 780–793, 2012.

[56] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, 2018.

[57] Y. Huang, Q. Wu, J. Xu, and Y. Zhong. Sbsgan: Suppression of inter-domain background shift for person re-identification. In *ICCV*, pages 9527–9536, 2019.

[58] L. Hubert and P. Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

[59] T. Isobe, D. Li, L. Tian, W. Chen, Y. Shan, and S. Wang. Towards discriminative representation learning for unsupervised person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[60] A. K. Jain and S. Z. Li. *Handbook of face recognition*, volume 1. Springer, 2011.

[61] Z. Ji, X. Zou, X. Lin, X. Liu, T. Huang, and S. Wu. An attention-driven two-stage clustering method for unsupervised person re-identification. In *European Conference on Computer Vision (ECCV)*, 2020.

[62] X. Jin, C. Lan, W. Zeng, and Z. Chen. Global distance-distributions separation for unsupervised person re-identification. In *European Conference on Computer Vision*, 2020.

[63] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Person re-identification by unsupervised l1 graph learning. In *ECCV*, pages 178–195, 2016.

[64] M. Kostinger, M. Hirzer, P. Wohlhart, and et al. Large scale metric learning from equivalence constraints. In *CVPR*, pages 2288–2295, 2012.

[65] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, pages 1097–1105, 2012.

[66] A. Krogh and J. Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4, 1991.

[67] D. Li, X. Wei, X. Hong, and Y. Gong. Infrared-visible cross-modal person re-identification with an x modality. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4610–4617, 2020.

[68] J. Li and S. Zhang. Joint visual and temporal consistency for unsupervised domain adaptive person re-identification. In *European Conference on Computer Vision*, pages 483–499. Springer, 2020.

[69] M. Li, X. Zhu, and S. Gong. Unsupervised person re-identification by deep learning tracklet association. In *ECCV*, pages 737–753, 2018.

[70] S. Li, J. Li, H. Tang, R. Qian, and W. Lin. Atrw: A benchmark for amur tiger re-identification in the wild. *arXiv preprint arXiv:1906.05586*, 2019.

[71] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, pages 152–159, 2014.

[72] Y.-J. Li, C.-S. Lin, Y.-B. Lin, and Y.-C. F. Wang. Cross-dataset person re-identification via unsupervised pose disentanglement and adaptation. In *ICCV*, pages 7919–7929, 2019.

[73] Y.-J. Li, C.-S. Lin, Y.-B. Lin, and Y.-C. F. Wang. Cross-dataset person re-identification via unsupervised pose disentanglement and adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[74] Y.-J. Li, F.-E. Yang, Y.-C. Liu, Y.-Y. Yeh, X. Du, and Y.-C. Frank Wang. Adaptation and re-identification network: An unsupervised deep transfer learning approach to person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[75] S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, pages 2197–2206, 2015.

[76] S. Liao and S. Z. Li. Efficient psd constrained asymmetric metric learning for person re-identification. In *ICCV*, pages 3685–3693, 2015.

[77] S. Liao and L. Shao. Interpretable and generalizable person re-identification with query-adaptive convolution and temporal lifting. In *European Conference on Computer Vision*, pages 456–474. Springer, 2020.

[78] S. Lin, H. Li, C. T. Li, and A. C. Kot. Multi-task mid-level feature alignment network for unsupervised cross-dataset person re-identification. In *29th British Machine Vision Conference, BMVC 2018*, 2018.

[79] Y. Lin, X. Dong, L. Zheng, Y. Yan, and Y. Yang. A bottom-up clustering approach to unsupervised person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[80] Y. Lin, Y. Wu, C. Yan, M. Xu, and Y. Yang. Unsupervised person re-identification via cross-camera similarity exploration. *IEEE Transactions on Image Processing*, 29:5481–5490, 2020.

[81] C. Liu, R. Zhang, and L. Guo. Part-pose guided amur tiger re-identification. In *IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[82] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang. Deep relative distance learning: Tell the difference between similar vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[83] J. Liu, Z.-J. Zha, D. Chen, R. Hong, and M. Wang. Adaptive transfer network for cross-domain person re-identification. In *CVPR*, pages 7202–7211, 2019.

[84] J. Liu, Z.-J. Zha, D. Chen, R. Hong, and M. Wang. Adaptive transfer network for cross-domain person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7202–7211, 2019.

[85] N. Liu, Q. Zhao, N. Zhang, X. Cheng, and J. Zhu. Pose-guided complementary features learning for amur tiger re-identification. In *IEEE/CVF international conference on computer vision workshops*, pages 0–0, 2019.

[86] X. Liu, W. Liu, T. Mei, and H. Ma. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *European conference on computer vision*, 2016.

[87] X. Liu and S. Zhang. Domain adaptive person re-identification via coupling optimization. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.

[88] Z. Liu, D. Wang, and H. Lu. Stepwise metric promotion for unsupervised video person re-identification. In *ICCV*, pages 2429–2438, 2017.

[89] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, 2018.

[90] C. Luo, C. Song, and Z. Zhang. Generalizing person re-identification by camera-aware invariance learning and cross-domain mixup. In *European Conference on Computer Vision (ECCV)*, 2020.

[91] C. Luo, C. Song, and Z. Zhang. Generalizing person re-identification by camera-aware invariance learning and cross-domain mixup. In *Computer*

*Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pages 224–241. Springer, 2020.

[92] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[93] J. Lv, W. Chen, Q. Li, and C. Yang. Unsupervised cross-dataset person re-identification by transfer learning of spatial-temporal patterns. In *CVPR*, pages 7948–7956, 2018.

[94] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato. Hierarchical gaussian descriptor for person re-identification. In *CVPR*, pages 1363–1372, 2016.

[95] L. McInnes, J. Healy, and S. Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017.

[96] D. Mekhazni, A. Bhuiyan, G. Ekladious, and E. Granger. Unsupervised domain adaptation in the dissimilarity space for person re-identification. In *European Conference on Computer Vision*, 2020.

[97] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, X. Yang, L. Zheng, A. Sharma, R. Chellappa, and P. Chakraborty. The 4th ai city challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[98] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 2013.

[99] H.-T. Nguyen and A. Caplier. Local patterns of gradients for face recognition. *IEEE Transactions on Information Forensics and Security*, 10(8):1739–1751, 2015.

[100] H.-T. Nguyen, N.-S. Vu, and A. Caplier. How far we can improve micro features based face recognition systems? In *2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 350–353, 2012.

[101] F. Pan, I. Shin, F. Rameau, S. Lee, and I. S. Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[102] H. Park, S. Lee, J. Lee, and B. Ham. Learning by aligning: Visible-infrared person re-identification using cross-modal correspondences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12046–12055, 2021.

[103] J. Peng, Y. Wang, H. Wang, Z. Zhang, X. Fu, and M. Wang. Unsupervised vehicle re-identification with progressive adaptation. *arXiv preprint arXiv:2006.11486*, 2020.

[104] P. Peng, T. Xiang, Y. Wang, M. Pontil, S. Gong, T. Huang, and Y. Tian. Unsupervised cross-dataset transfer learning for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1306–1315, 2016.

[105] X. Peng, Z. Huang, X. Sun, and K. Saenko. Domain agnostic learning with disentangled representations. In *International Conference on Machine Learning (ICML)*, pages 5102–5112, 2019.

[106] X. Peng, B. Usman, N. Kaushik, D. Wang, J. Hoffman, and K. Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[107] P. O. Pinheiro. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[108] L. Qi, L. Wang, J. Huo, L. Zhou, Y. Shi, and Y. Gao. A novel unsupervised camera-aware domain adaptation framework for person re-identification. In *ICCV*, pages 8080–8089, 2019.

[109] L. Qi, L. Wang, J. Huo, L. Zhou, Y. Shi, and Y. Gao. A novel unsupervised camera-aware domain adaptation framework for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[110] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[111] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision*, 2016.

[112] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015.

[113] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[114] A. Saporta, T.-H. Vu, M. Cord, and P. Pérez. Multi-target adversarial frameworks for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9072–9081, 2021.

[115] S. Schneider, G. W. Taylor, and S. C. Kremer. Similarity learning networks for animal individual re-identification-beyond the capabilities of a human observer. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 44–52, 2020.

[116] R. Shu, H. H. Bui, H. Narui, and S. Ermon. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*, 2018.

[117] C. Song, Y. Huang, W. Ouyang, and L. Wang. Mask-guided contrastive attention model for person re-identification. In *CVPR*, pages 1179–1188, 2018.

[118] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *CVPR*, pages 719–728, 2019.

[119] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 719–728, 2019.

[120] L. Song, C. Wang, L. Zhang, B. Du, Q. Zhang, C. Huang, and X. Wang. Unsupervised domain adaptive re-identification: Theory and practice. *Pattern Recognition*, 2020.

[121] M. Sugiyama, M. Krauledat, and K.-R. MÃžller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(May):985–1005, 2007.

[122] Y. Suh, J. Wang, S. Tang, T. Mei, and K. Mu Lee. Part-aligned bilinear representations for person re-identification. In *ECCV*, pages 402–419, 2018.

[123] X. Sun and L. Zheng. Dissecting person re-identification from the viewpoint of viewpoint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[124] H. Tang, Y. Zhao, and H. Lu. Unsupervised person re-identification with iterative self-supervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[125] A. Tarvainen and H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

[126] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[127] V. Vapnik. *Statistical learning theory*. Wiley, 1998.

[128] N.-S. Vu and A. Caplier. Illumination-robust face recognition using retina modeling. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 3289–3292, 2009.

[129] N.-S. Vu and A. Caplier. Enhanced patterns of oriented edge magnitudes for face recognition and image matching. *IEEE Transactions on Image Processing*, 21(3):1352–1365, 2012.

[130] D. Wang and S. Zhang. Unsupervised person re-identification via multi-label classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[131] G. Wang, T. Zhang, J. Cheng, S. Liu, Y. Yang, and Z. Hou. Rgb-infrared cross-modality person re-identification via joint pixel and feature alignment. In *ICCV*, pages 3623–3632, 2019.

[132] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin. Generalizing to unseen domains: A survey on domain generalization. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4627–4635, 2021.

[133] J. Wang, X. Zhu, S. Gong, and W. Li. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[134] M. Wang, B. Lai, J. Huang, X. Gong, and X.-S. Hua. Camera-aware proxies for unsupervised person re-identification. In *AAAI*, volume 2, page 4, 2021.

[135] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu. Shape and appearance context modeling. In *ICCV*, pages 1–8, 2007.

[136] X. Wang, R. Panda, M. Liu, Y. Wang, and et al. Exploiting global camera network constraints for unsupervised video person re-identification. *arXiv preprint arXiv:1908.10486*, 2019.

[137] Y. Wang, L. Wang, Y. You, X. Zou, V. Chen, S. Li, G. Huang, B. Hariharan, and K. Q. Weinberger. Resource aware person re-identification across multiple resolutions. In *CVPR*, pages 8042–8051, 2018.

[138] Z. Wang, J. Zhang, L. Zheng, Y. Liu, Y. Sun, Y. Li, and S. Wang. Cycas: Self-supervised cycle association for learning re-identifiable descriptions. In *European Conference on Computer Vision*, pages 72–88. Springer, 2020.

[139] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, pages 79–88, 2018.

[140] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[141] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person re-identification using kernel-based metric learning methods. In *ECCV*, pages 1–16, 2014.

[142] S. Xuan and S. Zhang. Intra-inter camera similarity for unsupervised person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[143] F. Yang, K. Li, Z. Zhong, Z. Luo, X. Sun, H. Cheng, X. Guo, F. Huang, R. Ji, and S. Li. Asymmetric co-teaching for unsupervised cross-domain person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[144] F. Yang, Z. Zhong, Z. Luo, Y. Cai, Y. Lin, S. Li, and N. Sebe. Joint noise-tolerant learning and meta camera shift adaptation for unsupervised person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[145] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li. Salient color names for person re-identification. In *ECCV*, pages 536–551, 2014.

[146] M. Ye, X. Lan, and P. C. Yuen. Robust anchor embedding for unsupervised video person re-identification in the wild. In *ECCV*, pages 170–186, 2018.

[147] M. Ye, A. J. Ma, L. Zheng, J. Li, and P. C. Yuen. Dynamic label graph matching for unsupervised video re-identification. In *ICCV*, pages 5142–5150, 2017.

[148] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[149] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014.

[150] K. You, X. Wang, M. Long, and M. Jordan. Towards accurate model selection in deep unsupervised domain adaptation. In *International Conference on Machine Learning*, 2019.

[151] H.-X. Yu, A. Wu, and W.-S. Zheng. Unsupervised person re-identification by deep asymmetric metric embedding. *IEEE TPAMI*, 2018.

[152] H.-X. Yu, W.-S. Zheng, A. Wu, X. Guo, S. Gong, and J.-H. Lai. Unsupervised person re-identification by soft multilabel learning. In *CVPR*, pages 2148–2157, 2019.

[153] H.-X. Yu, W.-S. Zheng, A. Wu, X. Guo, S. Gong, and J.-H. Lai. Unsupervised person re-identification by soft multilabel learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[154] M. Zając, K. Zolna, and S. Jastrzębski. Split batch normalization: Improving semi-supervised learning under domain shift. *arXiv preprint arXiv:1904.03515*, 2019.

[155] K. Zeng, M. Ning, Y. Wang, and Y. Guo. Hierarchical clustering with hard-batch triplet loss for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[156] Y. Zhai, S. Lu, Q. Ye, X. Shan, J. Chen, R. Ji, and Y. Tian. Ad-cluster: Augmented discriminative clustering for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[157] Y. Zhai, Q. Ye, S. Lu, M. Jia, R. Ji, and Y. Tian. Multiple expert brainstorming for domain adaptive person re-identification. In *Computer Vision–ECCV 2020: 16th European Conference*, 2020.

[158] X. Zhang, J. Cao, C. Shen, and M. You. Self-training with progressive augmentation for unsupervised cross-domain person re-identification. In *ICCV*, pages 8222–8231, 2019.

[159] X. Zhang, J. Cao, C. Shen, and M. You. Self-training with progressive augmentation for unsupervised cross-domain person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[160] X. Zhang, Y. Ge, Y. Qiao, and H. Li. Refining pseudo labels with clustering consensus over generations for unsupervised object re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[161] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu. Deep mutual learning. In *CVPR*, pages 4320–4328, 2018.

[162] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[163] F. Zhao, S. Liao, G.-S. Xie, J. Zhao, K. Zhang, and L. Shao. Unsupervised domain adaptation with noise resistible mutual-training for person re-identification. In *European Conference on Computer Vision (ECCV)*, 2020.

[164] R. Zhao, W. Ouyang, and X. Wang. Unsupervised salience learning for person re-identification. In *CVPR*, pages 3586–3593, 2013.

[165] F. Zheng, C. Deng, X. Sun, X. Jiang, X. Guo, Z. Yu, F. Huang, and R. Ji. Pyramidal person re-identification via multi-loss dynamic training. In *CVPR*, pages 8514–8522, 2019.

[166] K. Zheng, C. Lan, W. Zeng, Z. Zhang, and Z.-J. Zha. Exploiting sample uncertainty for domain adaptive person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[167] K. Zheng, W. Liu, L. He, T. Mei, J. Luo, and Z.-J. Zha. Group-aware label transfer for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[168] K. Zheng, W. Liu, L. He, T. Mei, J. Luo, and Z.-J. Zha. Group-aware label transfer for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5310–5319, 2021.

[169] L. Zheng, K. Idrissi, C. Garcia, S. Duffner, and A. Baskurt. Triangular similarity metric learning for face verification. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–7, 2015.

[170] L. Zheng, K. Idrissi, C. Garcia, S. Duffner, and A. Baskurt. Triangular similarity metric learning for face verification. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–7. IEEE, 2015.

[171] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, pages 1116–1124, 2015.

[172] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, 2015.

[173] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Y. Yang, and Q. Tian. Person re-identification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1367–1376, 2017.

[174] W.-S. Zheng, S. Gong, and T. Xiang. Person re-identification by probabilistic relative distance comparison. In *CVPR*, pages 649–656, 2011.

[175] Y. Zheng, S. Tang, G. Teng, Y. Ge, K. Liu, J. Qin, D. Qi, and D. Chen. Online pseudo label generation by hierarchical cluster dynamics for adaptive person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[176] Z. Zheng, L. Zheng, and Y. Yang. A discriminatively learned cnn embedding for person re-identification. *arXiv preprint arXiv:1611.05666*, 2016.

[177] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[178] Z. Zhong, L. Zheng, S. Li, and Y. Yang. Generalizing a person retrieval model hetero-and homogeneously. In *ECCV*, pages 172–188, 2018.

[179] Z. Zhong, L. Zheng, S. Li, and Y. Yang. Generalizing a person retrieval model hetero-and homogeneously. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[180] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*, pages 598–607, 2019.

[181] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[182] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2017.

[183] Y. Zou, X. Yang, Z. Yu, B. V. Kumar, and J. Kautz. Joint disentangling and adaptation for cross-domain person re-identification. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 87–104. Springer, 2020.

# List of Figures

# List of Tables

# Summary

**(FR)** La Re-Identification (re-ID) consiste en l'association d'observations de la même instance. En vision par ordinateur, la re-ID devient une tâche d'interprétation de l'image, pour laquelle les observations des différentes instances correspondent aux images d'une classe sémantique d'intérêt. La re-ID est au centre d'un grand nombre d'applications de la vision par ordinateur : la reconnaissance faciale, le suivi d'objets dans un réseau de caméras, la recherche par images (et d'images) dans une base de données... Devant le nombre croissant d'images collecter, et à interpréter, il devient nécessaire d'automatiser ces applications, et donc la re-ID.

La difficulté pour la re-ID automatique est de pouvoir extraire des images l'information pertinente relative à l'identité de l'instance. Cette information doit être suffisante afin de distinguer les différentes instances d'une classe sémantique d'intérêt, tout en étant suffisamment robuste aux différents facteurs de variabilité, afin de permettre le regroupement d'images même instance prise dans différentes conditions. L'apprentissage automatique supervisé, a permis d'inférer des caractéristiques discriminantes de l'identité pour la re-ID, à partir d'un grand nombre d'images d'instances dont les identités ont été annotées manuellement. Ces images sont modélisées comme des tirages d'une distribution de probabilité (ou domaine), supposée être la même que celle des images de test dans le cadre de l'apprentissage supervisée. Or, le domaine de test diffère couramment en pratique du domaine d'entraînement, si l'on souhaite déployer l'extracteur de caractéristiques dans un contexte différent : dans un lieu différent, avec des caméras différentes, ... Cependant, une différence des domaines causée par un changement de lieu, entraîne par exemple une chute drastique des performances de re-ID sur le domaine de test. Cela nécessite donc, dans le cadre de l'apprentissage supervisé, de collecter et d'annoter de nouvelles données, afin d'entraîner un nouvel extracteur de caractéristiques à chaque changement de lieu. Étant donné le coût important des annotations manuelles, et face à la nécessité pratique de résoudre ce problème de chute des performances de re-ID sur un nouveau domaine de test, cette thèse propose de s'intéresser à l'adaptabilité inter-domaines. L'adaptabilité inter-domaine consiste à transférer pour une tâche donnée, les connaissances pertinentes du domaine d'entraînement (domaine source) vers un domaine de test (domaine cible) différent. Pour cela, cette thèse s'intéresse plus particulièrement au cadre de l'adaptation de domaine non supervisée, qui suppose l'accès à des données annotées du domaine source, et des données non annotées du domaine cible. En effet, ce cadre de l'adaptation de domaine non supervisée reflète les contraintes pratiques du coût des annotations.

Cette thèse propose en premier lieu une analyse de l'état de l'art de l'adaptation de domaine non supervisée pour la re-ID. Cette analyse suggère que les méthodes par pseudo-labels semblent les plus prometteuses en termes de performance. La ligne directrice choisie pour cette thèse est donc d'exploiter le domaine source davantage afin d'améliorer ces approches par pseudo-labels qui semblent la sous-exploiter.

Une première contribution propose une approche basée sur l'intuition pour exploiter les données sources. L'idée est de montrer expérimentalement qu'il est possible d'améliorer la performance de re-ID inter-domaines en apprenant un modèle par pseudo-labels utilisant les données sources étiquetées, en plus des données cible pseudo-étiquetées. L'interprétation des expériences menées dans ce chapitre, indique que pour bénéficier des données sources, il est nécessaire de limiter l'impact du biais du domaine source sur le modèle pendant le processus d'apprentissage.

L'approche expérimentale montre des limites pour déterminer et comprendre toutes les bonnes pratiques générales pour bénéficier systématiquement de la connaissance source. Une deuxième contribution propose donc une approche théorique à ce problème. Celle-ci nous permet de déduire les bonnes pratiques générales à mettre en œuvre dans une méthode par pseudo-labels afin de bénéficier systématiquement de la connaissance de la source. De plus, cela permet de mieux comprendre le rôle de cette connaissance de la source dans l'amélioration des performances inter-domaines. Cependant, les approches par pseudo-labels, dans le contexte de l'adaptation de domaine non supervisée, sont toujours confrontées à un problème majeur qui limite la fiabilité de leurs performances inter-domaines en pratique : la sensibilité des performances inter-domaines aux hyperparamètres de clustering. Ce problème est renforcé par la difficulté de les sélectionner sans label pour les données du domaine cible dans le contexte de l'adaptation de domaine non supervisée.

Une troisième contribution propose donc HyPASS, une méthode de sélection automatique de ces hyperparamètres, en utilisant les données sources étiquetées. HyPASS est motivé par des développements théoriques. Il est démontré que le guidage par la source, ainsi que l'alignement conditionnel des similarités de caractéristiques entre les domaines, sont essentiels à la sélection automatique des hyperparamètres de clustering à l'aide d'un ensemble de validation de de données sources étiquetées. Diverses expériences montrent l'efficacité de HyPASS pour améliorer les performances inter-domaines, tout en offrant une plus grande fiabilité de ces performances, par rapport à un choix empirique de ces hyperparamètres.

Enfin, une conclusion met ce travail de thèse en perspective par rapport aux avancées des directions de recherche alternatives qui peuvent traiter le problème de la re-ID inter-domaines (l'apprentissage non supervisé et la généralisation de domaines pour la re-ID), ce qui permet de proposer des directions de recherche futures.

**(EN)** Re-Identification (re-ID) is the association of observations of the same instance. In computer vision, re-ID becomes an image interpretation task, for which the observations of different instances correspond to images of a semantic class of interest. Re-ID is at the heart of many computer vision applications: facial recognition, object tracking in a camera network, image (and image) search in a database... With the increasing number of images to collect and interpret, automating these applications and thus the re-ID becomes necessary.

The automatic re-ID's difficulty is extracting the relevant information about the identity of the instance from the images. This information must be sufficient to distinguish between different instances of a semantic class of interest while being sufficiently robust to different variability factors to allow clustering of the same images taken under different conditions. Supervised machine learning was used to infer discriminative identity features for re-ID from many images of instances whose identities were manually annotated. These images are modeled as draws from a probability distribution (or domain), assumed to be the same as the test images in supervised learning. However, the test domain commonly differs in practice from the training domain if one wishes to deploy the feature extractor in a different context: in a different location, with different cameras, ... However, a simple difference in domains caused by a change of location leads to a drastic drop in re-ID performance on the test domain. In the context of supervised learning, this requires collecting and annotating new data to train a new feature extractor at each location change. Given the high cost of manual annotations, and the practical need to solve this problem of falling re-ID performance on a new test domain, this thesis proposes to focus on cross-domain adaptability. Cross-domain adaptability consists in transferring, for a given task, the relevant knowledge from the training domain (source domain) to a different test domain (target domain). To this end, this thesis focuses on the framework of unsupervised domain adaptation. This framework assumes access to annotated data from the source domain and unannotated data from the target domain. Indeed, this framework of unsupervised domain adaptation reflects the practical constraints of the cost of annotations.

This thesis first proposes an analysis of unsupervised domain adaptation's state of the art for re-ID. This analysis suggests that pseudo-label methods seem to be the most promising in terms of performance. Therefore, the guideline chosen for this thesis is to exploit the source domain more to improve these pseudo-labels approaches, which under-exploit it.

A first contribution proposes an intuition-based approach to exploit the source data.

The idea is to show experimentally that it is possible to improve the performance of cross-domain re-ID by learning a pseudo-label model using the labeled source data, in addition to the pseudo-labeled target data. The interpretation of the experiments conducted in this chapter indicates that to benefit from the source data, it is necessary to limit the impact of the source domain bias on the model during the learning process.

The experimental approach shows limitations in systematically determining and understanding all the general good practices to benefit from the source knowledge. A second contribution, therefore, proposes a theoretical approach to this problem. This one allows us to deduce the general good practices to be implemented in a pseudo-labeling method in order to benefit from the source knowledge systematically. Moreover, it allows us to better understand the role of this knowledge of the source in the improvement of inter-domain performances. However, pseudo-labeling approaches, in the context of unsupervised domain adaptation, still face a significant problem that limits the reliability of their inter-domain performance in practice: the sensitivity of inter-domain performance to clustering hyperparameters. This problem is reinforced by the difficulty of selecting them without labels for the target domain data in unsupervised domain adaptation.

A third contribution, therefore, proposes HyPASS, a method for automatically selecting these hyperparameters using labeled source data. HyPASS is motivated by theoretical developments. It is shown that source guidance and conditional alignment of feature similarities between domains are essential for the automatic selection of clustering hyperparameters using a validation set of labeled source data. Various experiments show the effectiveness of HyPASS in improving inter-domain performance while providing more excellent reliability of this performance compared to an empirical selection of these hyperparameters.

Finally, a conclusion puts this thesis work in perspective concerning advances in alternative research directions that can address the problem of cross-domain re-ID (unsupervised learning and domain generalization for re-ID), thus proposing future research directions.