Beyond Dataset Bias: Multi-task Unaligned Shared Knowledge Transfer

Tatiana Tommasi[1,2], Novi Quadrianto[3], Barbara Caputo[1],
Christoph H. Lampert[4]

[1]Idiap Research Institute, Martigny, CH
[2] École Polytechnique Fédérale de Lausanne, CH
[3] University of Cambridge, UK
[4] IST Austria (Institute of Science and Technology Austria), Klosterneuburg, AT

Abstract. Many visual datasets are traditionally used to analyze the performance of different learning techniques. The evaluation is usually done within each dataset, therefore it is questionable if such results are a reliable indicator of true generalization ability. We propose here an algorithm to exploit the existing data resources when learning on a new multiclass problem. Our main idea is to identify an image representation that decomposes orthogonally into two subspaces: a part specific to each dataset, and a part generic to, and therefore shared between, all the considered source sets. This allows us to use the generic representation as un-biased reference knowledge for a novel classification task. By casting the method in the multi-view setting, we also make it possible to use different features for different databases. We call the algorithm MUST, Multitask Unaligned Shared knowledge Transfer. Through extensive experiments on five public datasets, we show that MUST consistently improves the cross-datasets generalization performance.

1 Introduction

The long standing ambition of the visual recognition community has been to enable artificial visual systems to recognize not only specific instances of a category, such as *my car*, but *cars* in general. Many visual databases (e.g. Caltech 101 [1], PASCAL VOC [2], Animals with Attributes [3], ImageNet [4]) have been created to support such quest. However, recent studies [5, 6] have questioned if the results obtained so far are a reliable indicator of real generalization abilities. Indeed, it seems that high performance on a data collection often does not reflect on the ability to classify correctly the same classes, imaged in another dataset.

One of the main reasons behind this problem is the data selection bias [5]: images contained in two databases under the same category label can represent instead different related subcategories, e.g. in ImageNet the class "car" has a strong preference for race cars. Conversely (category label bias [5]), it might happen that different labels are used for the same type of object, e.g. the class "dog" in PASCAL presents images of "collie" and "dalmatian" breed dogs that correspond instead to two separate classes in Animals with Attributes.

When looking at the disappointing cross-dataset generalization results reported in [5] keeping in mind the biases described above, one could formulate an hypothesis: a classifier trained on a specific dataset learns, for each object class, a model containing some generic knowledge about the semantic categorical problem, and some specific knowledge about the bias contained into that dataset. For example for the object category "car", a classifier trained on ImageNet would learn a racing car model. Still, the specific ability to classify correctly race cars implies having some knowledge about the general category car.

Issues arise even when focusing only on common classes across multiple existing datasets, as their label name is not sufficient to select and align them. It is necessary to inspect visually their content or use a pre-defined hierarchical ontology (like Wordnet [7]). Moreover, analyzing one class at a time implies the definition of binary problems where the negative class is obtained by sampling from the remaining set of classes, specific to each database. Thus, the definition of *what an object is not* is intrinsically biased (negative bias [5]).

Here we propose a method to overcome these issues. We exploit existing visual datasets preserving their multiclass structure and relying on the fact that they are many: each of them presents specific characteristics, but all together they cover different nuances of the real world. As the data are not uniformly distributed [8], it often happens that some classes overlap across the datasets, giving us the possibility to learn on them decoupling explicitly the generic and specific knowledge. The common information can then be used on any new multiclass problem. Along this line our main contributions are (1) we generalize the dataset bias problem presented in [5] to multiclass and to heterogeneous features: often the biases are induced by a specific research focus which turns in some features being more appropriate for some databases; (2) we introduce our Multi-task Unaligned Shared knowledge Transfer (MUST) algorithm that learns jointly shared and private knowledge from multiple datasets, and then transfers the common information when training on a new dataset. By casting the problem within the multi-view learning setting, we are able to use, for each database, features previously proposed, pre-computed and publicly available for download; (3) we propose for the first time a leave-one-dataset-out experimental setup over five existing datasets that can be considered a valid test bed for any cross-dataset generalization method.

In the rest of the paper we define our learning problem, and review related work (Section 2). We then describe the model (Section 3) and its extension to the multi-view setting (Section 4). Experiments are presented in Section 5. We conclude the paper with an overall discussion.

2 Problem Statement and Related Work

We formalize here the problem of learning a classifier on a target set when many source sets are available, in the hypothesis of a distribution mismatch between the target and the sources, and across the sources.

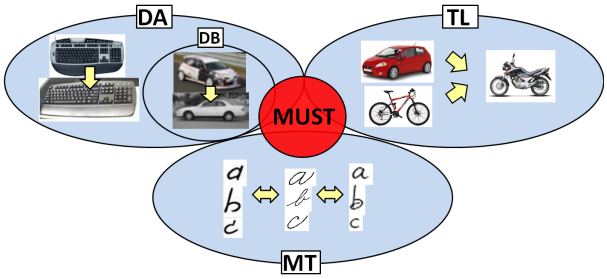

Fig. 1. Examples of existing approaches to the distribution mismatch problem. **DA:** adapt from Amazon keyboards to images of keyboards acquired in a specific office. **DB**: the difference between ImageNet cars and Caltech 101 cars is shown by the bad results obtained when learning on the first and testing on the second. **TL**: extract information from a car and a bicycle and use it when learning motorbikes from few examples. **MT**: learn to classify letters from the handwriting of many subjects. Our MUST algorithm partially overlap with all the described methods, filling in the empty space among them.

Let's indicate with $X \in \mathcal{X}$ the data and with $Y \in \mathcal{Y}$ the corresponding labels, where $\mathcal{X}$ and $\mathcal{Y}$ specify respectively the feature and the label space. We call *domain* $D = \{\mathcal{X}, P(X)\}$ the couple of feature space and marginal distribution on the data, while a *task* $T = \{\mathcal{Y}, P(Y|X)\}$ is the couple of label space and prediction function written in probabilistic terms. Depending on (a) what gives rise to the distribution mismatch in terms of domain and task relations, and (b) if the learning process is symmetric or asymmetric over the multiple data sets, it is possible to consider different solutions to specific subparts of the general problem. We describe them below, giving corresponding examples in Figure 1.

Domain Adaptation (DA) aims at solving the learning problem on a target domain D^t exploiting information from a source domain D^s, when both the domains and the corresponding tasks T^s, T^t are not the same. In particular, the tasks have identical label sets $\mathcal{Y}^s = \mathcal{Y}^t$ but with slightly different conditional distributions $P^s(Y|X) \sim P^t(Y|X)$. The domains are different in terms of marginal data distribution $P^s(X) \neq P^t(X)$, and/or in feature spaces $\mathcal{X}^s \neq \mathcal{X}^t$.

DA is well studied in machine learning [9, 10], speech and language processing [11, 12] and more recently in computer vision, both in the semi-supervised [13] and unsupervised settings [14, 15]. In case of multiple sources either the information extracted is averaged over all of them [14], or specific methods are proposed to select the best source [15]. The particular problem of domain shift across common classes in different datasets has been identified with the name of Dataset Bias (DB) [5].

Transfer Learning (TL) focuses on the possibility to pass useful knowledge from a source task to a target task with different label sets $\mathcal{Y}_s \neq \mathcal{Y}_t$, when the corresponding domains are not the same but the marginal distributions of data are related $P^s(X) \sim P^t(X)$. TL has been widely studied in the binary setting across couples of categories [16] and recently has been extended to multiclass problems [17]. One of the main issues here is how to evaluate the task relatedness

before transferring, on the basis of only few available labeled samples in the target and eventually multiple sources.

Multi-Task Learning (MT) aims at learning jointly over N available sets, leading to a symmetric share of information. This is particularly useful when each task has few data. The multi-task framework supposes that all the sets share the same feature space $\mathcal{X}^i = \mathcal{X}^j$ but present slightly different domains $P^i(X) \sim P^j(X)$ for $i, j \in \{1, \ldots, N\}$. Traditionally, one either assume that the set of labels for all the tasks are the same ($\mathcal{Y}^s = \mathcal{Y}^t$) or that it is possible to access to an oracle mapping function $\mathcal{Y}^s \mapsto \mathcal{Y}^t$ that aligns the classes. Many techniques for MT have been published in machine learning [18, 19] with some applications in computer vision [20, 21]. Most of the works suppose multiple binary tasks and only few attempts has been done in the multiclass case without label correspondences [22, 23].

Our MUST algorithm fits in the general setting of all these approaches, while covering issues orthogonal across all of them. We are interested in multiple sources and a single target with domain shift and partially overlapping label sets: $\mathcal{Y}^i \cap \mathcal{Y}^j \neq \emptyset$ for all $(i, j) \in \{1, \ldots, N\}$. The difference in the domains can be caused by both $P^i(X) \neq P^j(X)$ and $\mathcal{X}^i \neq \mathcal{X}^j$. The aim is to extract general information from all the sources (in multi-task fashion) and to use it when learning on a new target with a general advantage both on the known categories (as in domain adaptation) and on new ones (as in transfer learning). With respect to classical multi-task learning, we break the symmetry adding a transfer part to a target problem. At the same time, we overcome the transfer learning problem of evaluating the task relatedness leveraging on the possibility to extract a common useful knowledge from multiple sources. Finally, we go beyond domain adaptation which does not cover the case of completely new classes in the target task. Moreover, considering multiple sources (with eventually different features) we show that the hypothesis of relying on a flat average knowledge is not helpful in the case of tasks with partially overlapping label sets.

We pursue our goal by defining a method that allows us to exploit existing visual resources with a *minimal effort*: (1) we do not need to know explicitly which classes are present in each task and therefore, no manual alignment is necessary, (2) we do not need to keep the source data when learning on the target, (3) we leverage over multiple sources regardless to their feature space. MUST is inspired by recent research on finding shared and private projections [24, 25] for problems where multiple modalities or multiple views of the same data are available. This notion was also exploited in the context of Multi-Task learning in [23].

Recently the dataset bias problem has been explored in [26]. The proposed approach is based on the combination of a specific and a common discriminative model across several tasks, following the same idea of the original multi-task SVM [18]. The novelty is in the fact that the common model, apart from sharing information, is constrained to perform well on any task on its own. By using SVM, this strategy results intrinsically limited to binary problems: any SVM

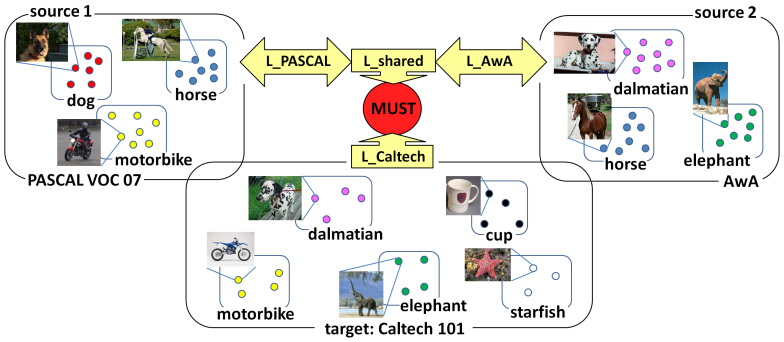

Fig. 2. Schematic representation of the MUST algorithm: shared and private information are extracted from two existing datasets. The shared knowledge is then transferred to solve a new multiclass problem on a different dataset. Notice that no explicit alignment is requested between "dalmatian" and "dog" classes.

multiclass solution considers one model for each class and this would ask for class alignment.

3 The Model

Starting from multiple visual object datasets, our goal is to learn a projection function that maps the data points into one shared and several private latent spaces with an *orthogonality* constraint between them. We can then transfer the knowledge encoded in the shared space to a new dataset and use the available training samples to learn only the remaining private orthogonal part (see Figure 2). The new problem will benefit from this approach only if the shared space captures non-dataset-specific information which we will call *common sense*.

More formally, we are given N sets of m_n observed data points, $\mathcal{D}_n = \{(x_1^n, y_1^n), \ldots, (x_{m_n}^n, y_{m_n}^n)\} \subset \mathcal{X}^n \times \mathcal{Y}^n$ for $n = 1, \ldots, N$. Here we use $\mathcal{X}^n$ and $\mathcal{Y}^n$ to denote the input space and output space of the n-th dataset. For the purpose of explaining the key idea, we assume that the same representation is used for all the datasets, $\mathcal{X}^n = \mathbb{R}^d$ for all n. We further require some overlap in the output spaces, i.e. $\mathcal{Y}^i \cap \mathcal{Y}^j \neq \emptyset$ for all $(i, j) \in \{1, \ldots, N\}$. The existence of such partial superposition in the label sets allows to introduce the notion of common sense as generic knowledge among the tasks. It is important to underline that we want an approach which does not require explicit label correspondences among datasets, and we are interested in models that do not build those correspondences as an intermediate learning step. We seek functions

$$g_n : \mathbb{R}^d \to \mathbb{R}^D \quad \text{for} \quad n = 1, \ldots, N, \tag{1}$$

which project the original space into a novel one with potentially much smaller dimension $D \ll d$. We assume a *linear* parametrization of the functions and an *additive* model for the shared-private spaces. Thus, the projection functions

admit the following form $g_n(x_i^n) := (L_n + L_s)\phi(x_i^n)$ for H basis functions[1] $\{\phi_h(x_i)\}_{h=1}^H$, a private projection matrix for the n-th dataset $L_n \in \mathbb{R}^{D\times H}$, and a shared projection matrix $L_s \in \mathbb{R}^{D\times H}$. We learn those projection matrices based on the *folk-wisdom* principle [28, 27, 29] of pulling objects or data samples together if they are of the same type (keeping your friends close), and pushing them apart if they are not (keeping your enemies far away). This principle is formalized by the regularized risk functional described in the following Section.

3.1 Regularized Risk Functional

We want to learn a transformation over the data by minimizing a function which penalizes large distances between samples of the same class, and small distances between samples with non-matching class labels. We assume that for each sample, it is possible to identify a set of genuine neighbors or friends. The notation $i \sim j$ is used to indicate that x_i and x_j are friends as belonging to the same class, and the notation $i \not\sim l$ describes that x_i and x_l are enemies as associated to different class labels. Our optimization problem has the following form:

$$\begin{aligned}\min_{\substack{L_s\\ L_{1,\dots,N}}} & \underbrace{\sum_{n=1}^{N}\sum_{i\sim j} d_n^2(x_i^n, x_j^n) + \sum_{\substack{i\sim j\\ i\not\sim l}} \max(0, 1 + d_n^2(x_i^n, x_j^n) - d_n^2(x_i^n, x_l^n))}_{\text{Loss}(\cdot)} \\ & + \eta\Omega(L_n) + \gamma\Omega(L_s) \qquad (2)\\ \text{subject to } & L_s^\top L_n = 0 \quad \text{for all } n = 1,\dots,N,\end{aligned}$$

where $d_n^2(x_i^n, x_j^n) := \left\|(L_n + L_s)(\phi(x_i^n) - \phi(x_j^n))\right\|_{\ell_2}^2$ is the squared distance in the projected space. In (2), Loss$(\cdot)$ is the loss function, $\Omega(\cdot)$ is a regularizer on the projection matrices, and the trade-off variables η and λ control the relative influence of loss and regularization terms. For $\Omega(\cdot)$, one typically chooses the ℓ_2 norm, or the ℓ_1 norm if one wants to induce sparsity in the projection matrices. The loss function consists of two terms: the first requires small distances among friend samples, while the second asks that the distance between each sample and its enemies is a unit greater than the corresponding distance to the friends. Finally, the constraints ensure that the inferred shared space is orthogonal to each of the private spaces.

Given a new dataset of m_t observed data points $\mathcal{D}_t = \{(x_1^t, y_1^t), \dots, (x_{m_t}^t, y_{m_t}^t)\} \subset \mathbb{R}^d \times \mathcal{Y}^t$ with $\mathcal{Y}^t \cap (\bigcup_{n=1,\dots,N} \mathcal{Y}^n) \neq \emptyset$ we want to learn its specific representation while enforcing it to be orthogonal to the common sense obtained from the previous N datasets. This corresponds to finding a private projection matrix L_t *given* the shared projection matrix L_s, and can be expressed with the following optimization problem:

[1] We use $\phi(x_i)$ to indicate the possibility of non-linear mapping applied on the original feature vector x_i. The full method might be kernelized by building on [27].

Algorithm 1 MUST

Input N source datasets $\mathcal{D}_n = \{(x_1^n, y_1^n), \ldots, (x_{m_n}^n, y_{m_n}^n)\} \subset \mathbb{R}^d \times \mathcal{Y}^n$
Input a target dataset $\mathcal{D}_t = \{(x_1^t, y_1^t), \ldots, (x_{m_t}^t, y_{m_t}^t)\} \subset \mathbb{R}^d \times \mathcal{Y}^t$
`Solve` optimization problem in (2) for shared L_s and private $L_{1,\ldots,N}$
`Transfer` the common sense as captured by L_s to a new dataset $\mathcal{D}_t$
Given L_s, `solve` optimization problems in (3) for private L_t
Output L_t

$$\min_{L_t} \sum_{i \sim j} d_t^2(x_i^t, x_j^t) + \sum_{\substack{i \sim j \\ i \not\sim l}} \max(0, 1 + d_t^2(x_i^t, x_j^t) - d_t^2(x_i^t, x_l^t)) + \eta \Omega(L_t) \quad (3)$$

$$\text{subject to } L_s^\top L_t = 0,$$

where $d_t^2(x_i^t, x_j^t) := \left\| (L_t + L_s)(\phi(x_i^t) - \phi(x_j^t)) \right\|_{\ell_2}^2$. Intuitively, whenever the common sense knowledge given by L_s is sufficient to enforce the folk-wisdom principle, there is no penalty incurred in (3). The learning capacity of the private projection matrix L_t can thus be focused on those hard cases specific to this new dataset. In the following Section, we go on describing the methods to optimize problems (2) and (3).

3.2 Optimization

The optimization problem (2) (and (3) likewise) is non-convex with respect to the projection matrices $L_s, L_1, \ldots, L_N$, thus it is hard to optimize. However, [27] and more recently [23] presented two ideas to turn the problem in (2) – excluding the orthogonality constraints – into a convex optimization problem, namely, a semi-definite programming. The first idea is to replace the second term of the loss function, with a soft margin constraint. This is achieved by introducing a non-negative slack variable for every pair of friends and enemies ξ_{ijl} such that $d_n^2(x_i^n, x_l^n) - d_n^2(x_i^n, x_j^n) \geq 1 - \xi_{ijl}$. This will essentially allow the distance between samples and their enemies to be less than a unit greater than the distance with their friends. To avoid this behavior for occurring often, there is a budget on the slack variables $\sum_{\substack{i \sim j \\ i \not\sim l}} \xi_{ijl}$ that needs to be minimized. The second intuition is to substitute the optimization over the projection matrix L with the optimization over the corresponding metric $M := L^\top L$, therefore imposing a semi-definite constraint on $M \succeq 0$.

Weinberger and Saul [27] described a convex solver based on alternating sub-gradient descent methods for the re-formulated problem. Recently, Kleiner, Rahimi, and Jordan [30] devised an approach to solve SDPs by repeatedly solving randomly generated optimization problems over two-dimensional subcones of the PSD cone. This approach produces only approximate solutions due to randomization, but it scales to number of samples orders of magnitude larger than have previously been possible. Here, we show that the same solvers can still be used for our constrained problem as the linearity and additive model

assumptions allow us to write

$$d_n^2(x_i^n, x_j^n) = \left\|(L_n + L_s)(\phi(x_i^n) - \phi(x_j^n))\right\|_{\ell_2}^2 \quad (4)$$

$$= \left\|L_n(\phi(x_i^n) - \phi(x_j^n))\right\|_{\ell_2}^2 + \left\|L_s(\phi(x_i^n) - \phi(x_j^n))\right\|_{\ell_2}^2 \ , \quad (5)$$

and its analogous for $d_t^2(x_i^t, x_j^t)$. The last equality follows directly from our orthogonality assumptions. Note that $\left\|L_s(\phi(x_i^t) - \phi(x_j^t))\right\|_{\ell_2}^2$ is fixed for each set of neighbors and thus can be pre-computed. In this paper, we use the solver presented in [27]. The full method MUST is summarized in Algorithm 1.

4 Multi-View on Multiple Datasets

We now consider the case where each of the given N datasets lies in its own feature space, that is $\mathcal{X}^n = \mathbb{R}^{d_n}$ for $n = 1, \ldots, N$. This setting easily appears since most of the visual datasets are released together with their own pre-extracted features. For this multi-view problem, we seek additional projection functions $f_n : \mathbb{R}^{d_n} \to \mathbb{R}^d$ that map all inputs from different databases to an intermediate $\mathbb{R}^d$ space in addition to finding the shared and private metrics. We assume a linear parametrization for the multi-view functions $f_n := W_n x_i^n$ where $W_n \in \mathbb{R}^{d \times d_n}$ is the multi-view projection matrix for the n-th dataset. Our multi-view distance function with the orthogonality constraint between shared and private spaces made explicit is now:

$$\hat{d}_n^2(x_i^n, x_j^n) = (W_n(\phi(x_i^n) - \phi(x_j^n)))^\top (L_n^\top L_n + L_s^\top L_s)(W_n(\phi(x_i^n) - \phi(x_j^n))) \quad (6)$$

$$= \mathrm{trace}(M_s W_n v_{ij}^n v_{ij}^{n,\top} W_n^\top) + \mathrm{trace}(M_n W_n v_{ij}^n v_{ij}^{n,\top} W_n^\top)$$

$$\text{with } M_s \succeq 0 \text{ and } M_n \succeq 0,$$

where $v_{ij}^n = (\phi(x_i^n) - \phi(x_j^n))$. We use the above distance function as a drop-in replacement to the objective function in (2). Thus the optimization problem will be over the multi-view projection matrices W_ns and over the metrics M_s , M_ns. Similarly to the single-view case, given a new dataset, we will solve the optimization problem in (3), but now an additional projection matrix $f_t : \mathbb{R}^{d_t} \to \mathbb{R}^d$ that bring the new datasets to the same intermediate space of the old training datasets has also to be found.

Optimization. The optimization problem in (2) with the modified distance function $\hat{d}_n^2(x_i^n, x_j^n)$ is convex with respect to the metrics given all the multi-view projection matrices W_ns and is non-convex with respect to the multi-view projection matrices given the shared and private metrics M_s and M_n. We pursue an alternating approach: fix all the multi-view projection matrices and solve the shared and private metrics M_s and M_n with [27]; subsequently, fix the metrics and optimize all the multi-view projection matrices W_ns with fast sub-gradient descent algorithm. In this paper, we use nonsmooth BFGS [31]. This procedure is repeated until a certain number of alternating steps is reached. The Multi-View (MUST-MV) version of our method is summarized in Algorithm 2.

Algorithm 2 MUST-MV

Input N source datasets $\mathcal{D}_n = \{(x_1^n, y_1^n), \ldots, (x_{m_n}^n, y_{m_n}^n)\} \subset \mathbb{R}^{d_n} \times \mathcal{Y}^n$
Input a target dataset $\mathcal{D}_t = \{(x_1^t, y_1^t), \ldots, (x_{m_t}^t, y_{m_t}^t)\} \subset \mathbb{R}^{d_t} \times \mathcal{Y}^t$
Input number of alternations A
`Initialize` $W_n^{d_n} = W_n^{\text{PCA}}\ \forall\ n = 1, \ldots, N$
for $a = 1$ **to** A **do**
  `Solve` optimization problem in (2) for shared L_s and private $L_{1,\ldots,N}$
  `Solve` optimization problem in (2) for multi-view projections W_n
end for
`Initialize` $W_t^{d_t} = W_t^{\text{PCA}}$
`Transfer` the common sense as captured by L_s to a new dataset $\mathcal{D}_t$
for $a = 1$ **to** A **do**
  Given L_s, `solve` optimization problem in (3) for private L_t
  `Solve` optimization problem in (3) for multi-view projection W_t
end for
Output L_t, $W_t^{d_t} \in \mathbb{R}^{d_t \times d}$

5 Experiments

We present here two groups of experiments designed to study how MUST[2] performs on *cross-database* generalization problems both in the case with all sets having the same feature representation (single-view setting, Section 5.1) and when each of the datasets lies in its own feature space (multi-view setting, Section 5.2). To this purpose, we selected five visual object databases which are actively used in present computer vision research and have some partial overlapping in the label space: Caltech 101 [1] with 101 class labels, PASCAL VOC07 [2] with 20 class labels, MSRCORID [32] with 20 class labels, Animals with Attributes (AwA) [3] with 50 class labels, and CIFAR 100 [33] with 100 class labels. We applied a one-dataset-out strategy, extracting the general knowledge from four datasets and giving the chance to each database in turn to be used as a new problem.

5.1 Single-View setting

For the single view experiments we extracted Gist features [34] from the images converted to grayscale and ran metric learning on the sources with 15 genuine neighbors, both considering the multi-task approach (using [23]) and keeping the task separated (using [27]). We fixed the maximum number of enemies to a very high value (10^6), letting the algorithm almost free to find all the active neighbors belonging to a different class. A first set of experiments was run on a subset of the listed datasets described in Figure 3(top, left): each dataset has a partial class overlapping with the others and two completely new categories. Here for each source database we have randomly chosen 90/30/30 samples per

[2] The code for MUST containing all the scripts used for the experiments is available online `http://www.idiap.ch/~ttommasi/source_code_ACCV12.html`

Classes	Caltech 101	PASCAL VOC 07	MSRCORID	AwA	CIFAR 100
car	X	X	X		
elephant	X			X	X
bottle		X			X
butterfly	X				X
cloud			X		X
dalmatian	X			X	
horse		X		X	X
bicycle		X	X		
sheep		X	X	X	
motorbike	X	X			X
chandelier	X				
grand piano	X				
potted plant		X			
boat		X			
chimneys			X		
doors			X		
walrus				X	
giraffe				X	
mushrooms					X
wardrobe					X

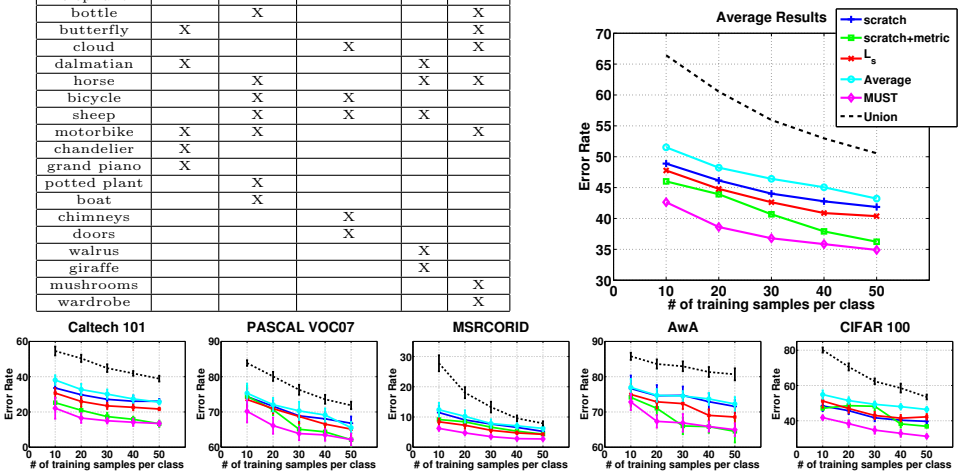

Fig. 3. Visual object classification across different datasets using the same feature. Top: (left) the table describing the experimental setup, (right) plot of the average results on five datasets. Bottom: separate results on each target dataset. Results over 10 repetitions for all methods except Union with 5 repetitions due to high computational demand.

class for training, validation and test. For the new target database we fixed a test set of 50 samples and considered an increasing number of available training samples $n = \{10, 20, 30, 40, 50\}$. Only for Caltech 101 we reduced the described sets respectively to 30/10/10 and we used 10 samples as test set, due to the smaller number of available data per class.

The performance of MUST is compared with four baselines, two corresponding to learning from scratch and two exploiting the shared knowledge with naïve transfer approaches:
scratch: we used the Identity as projection matrix (Euclidean metric);
scratch+metric: we learn a metric from the available new training data;
$\mathbf{L_s}$: the shared projection matrix L_s learned on multiple datasets is applied on the new one;
Average: projection matrices L_n learned separately on each database; their average is applied on the new dataset.

We can in principle combine all the samples from the visual datasets. It is already known [5] that this simple solution is not helpful against the dataset bias problem, moreover, apart from suffering for an explosion in the number of data, it requires an explicit class alignment procedure. However, as a reference to the results that could be obtained in this setting, we ran metric learning [27] on the **Union** of all the training samples. All the final classification are performed using k-Nearest Neighbor with $k = 15$ ($k = 8$ only for 10 available training samples).

Table 1. Error rate results obtained on the single-view experiments considering the whole datasets.

target	scratch+metric (%)	L_s(%)	Average (%)	MUST (%)
Caltech 101	65.69 ± 0.99	70.66 ± 1.87	75.35 ± 1.56	62.55 ± 1.08
Pascal VOC07	84.94 ± 3.14	84.50 ± 2.15	85.38 ± 3.42	80.66 ± 2.12
MSRCORID	45.80 ± 4.26	51.79 ± 2.73	52.59 ± 2.93	40.24 ± 3.11
AwA	94.02 ± 1.20	93.98 ± 0.84	94.24 ± 1.11	92.32 ± 1.18
CIFAR 100	90.91 ± 0.97	87.84 ± 1.06	92.76 ± 0.80	87.48 ± 0.78
overall	76.27	77.75	80.06	72.65

From the results in Figure 3 we can state that averaging over all the sources does not directly provide a good solution for the target problem. On the other hand, when only few training samples are available (10-20), by learning on them we get just slightly better performance w.r.t. using directly the general knowledge in L_s. However, when the number of samples increases, L_s is no more enough by itself to solve the learning problem on the new task. Finally, inferring the specific private knowledge on the new dataset and combining it with the shared common sense with our MUST algorithm *always* improves the average classification performance. By looking closely at the results on each new dataset, MUST mostly improves but *never degrades* the performance in comparison to not utilizing the available sources (scratch+metric in the plot).

We also performed a second set of experiments considering all the available classes in each dataset. We defined ten splits randomly extracting 20/10 train/test samples from each class of the target task dataset, 15 genuine neighbors and 100 enemies. Since the test set changes at each run, the standard deviations are only barely indicative. We evaluated the difference between MUST and scratch+metric separately for all the splits: the sign test [35] on the obtained output confirms that MUST significantly outperforms scratch+metric with $p \leq 0.05$. There is only one exception for AwA, the animals are highly confused among each other and in this case it is probably necessary to increase the number of enemies in the method to reach significant results.

5.2 Multi-View setting

In the multi-view setting we considered different features for each dataset. We used bag of words SIFT features[3] for Caltech 101, Hue color histogram[4] for PASCAL VOC07, the already calculated Gist for MSRCORID and PHOG features[5] for AwA. Finally we calculated PHOG for CIFAR but using different parameters with respect to the features used for AwA. We ran the experiments on the same data subset described above (Figure 3(top, left)): we applied PCA separately on the multiple tasks to project all of them in the same dimensional space with $D = \{10, 50\}$ before running the metric learning process to define the shared knowledge. On the novel dataset, we can again use PCA and proceed

[3] From `http://www.vision.ee.ethz.ch/~pgehler/projects/iccv09/`
[4] DenseHueV3H1 from `http://lear.inrialpes.fr/people/guillaumin/data.php`
[5] From `http://attributes.kyb.tuebingen.mpg.de/`

with MUST to learn the specific metric, or we can activate the optimization for the projection matrix W. We consider the first approach as a reference baseline and compare with MUST-MV.

The number of genuine neighbors and enemies for these experiments are fixed to 3 to infer the general and specific knowledge on each task and to 5 for learning the multi-view projections. These choices are done on the basis of two considerations. First, we want a good balance between computational cost and accuracy performance. Further, we aim to put a little more emphasis on retaining dataset-specific characteristics before inferring the shared knowledge in successive iterations. The last point lead us also to observe that for the multi-view problem, it is beneficial to have a dataset specific constant in (2) and (3) when enforcing the large difference between friends and enemies. Thus we substituted the value 1 in the second term of the loss function with the median of the squared pairwise distances in each dataset own feature space.

The results reported in Figure 4 show that on average MUST-MV is more suitable for the multi-view problem than the original MUST. Looking at the single target results, the advantage given by learning the projection matrix W_t is more evident the smaller is the dimension D. We also notice that MUST-MV performs always better (or at least equal) than MUST with one only exception when CIFAR 100 is used as target task with $D = 50$. We believe that in this particular case the combination of general and specific knowledge should be better weighted giving more importance to the common sense. This explanation is corroborated by the single-view CIFAR 100 results (Figure 3, bottom right) that show an initial abnormal increasing behavior for the scratch+metric baseline when the number of available training samples grows, while exploiting the common knowledge together with the specific one we get the best results.

6 Discussion and Conclusion

We presented here our MUST algorithm that decomposes multiple datasets into two orthogonal subspaces: one is specific to each dataset and the other is shared between all of them. Then the common information is transferred to help on a new task. On average, MUST *always* demonstrates cross-dataset generalization, assessed via a one-dataset-out strategy. We stress that the aim of our work was not to achieve the next state of the art accuracy on any of the considered databases, but rather to show that, in spite of the bias afflicting each of them, they do all carry a useful knowledge which is learnable and exploitable, significantly improving the generalization ability of a learning system.

By relying on metric learning and using a formulation similar to [27], MUST benefits of a max-margin framework analogous to that of SVM, but overcomes the class alignment limit of the SVM multiclass models. Moreover, the general and specific metrics produced by MUST can be used afterwords by any approach that requires distance computation among samples, including kernel methods.

Besides showing encouraging results, we have clearly only touched the surface of possibilities to be explored.

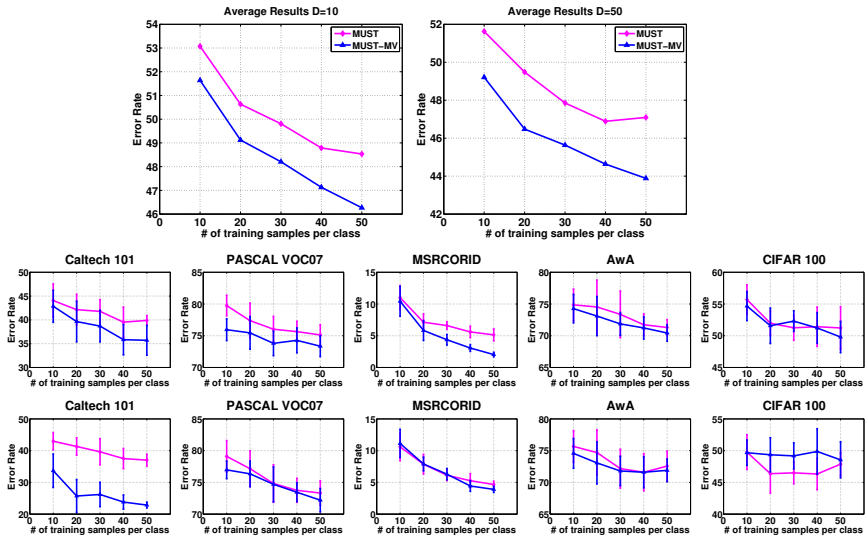

Fig. 4. Top: average error rate results on the five datasets considering the projection of all the different features to a space of dimension D=10,50 (left,right). Middle: separate results on each target datasets over 10 repetitions for D=10. Bottom: separate results on each target datasets over 10 repetitions for D=50. All the reported error rates for MUST-MV correspond to the best results obtained over the multiple iterations of the alternating optimization process.

Acknowledgements. This work was supported by the PASCAL 2 Network of Excellence (TT) and by the Newton International Fellowship (NQ).

References

1. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Comput. Vis. Image Underst. **106** (2007) 59–70
2. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. (http://www.pascal-network.org/challenges/VOC/)
3. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between class attribute transfer. In: CVPR. (2009)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR. (2009)
5. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: CVPR. (2011)
6. Perronnin, F., Sánchez, J., Liu, Y.: Large-scale image categorization with explicit data embedding. In: CVPR. (2010)
7. Stark, M.M., Riesenfeld, R.F.: Wordnet: An electronic lexical database. In: Eurographics Workshop on Rendering, MIT Press (1998)
8. Salakhutdinov, R., Torralba, A., Tenenbaum, J.: Learning to Share Visual Appearance for Multiclass Object Detection. In: CVPR. (2011)

9. Blitzer, J., Crammer, K., Kulesza, A., Pereira, O., Wortman, J.: Learning bounds for domain adaptation. In: NIPS. (2008)
10. Ben-david, S., Blitzer, J., Crammer, K., Sokolova, P.M.: Analysis of representations for domain adaptation. In: NIPS. (2007)
11. Blitzer, J., McDonald, R., Pereira, F.: Domain adaptation with structural correspondence learning. In: EMNLP. (2006)
12. Daumé III, H.: Frustratingly easy domain adaptation. In: ACL. (2007)
13. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: ECCV. (2010)
14. Gopalan, R., Li, R., Chellappa, R.: Domain adaptation for object recognition: An unsupervised approach. In: ICCV. (2011)
15. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: CVPR. (2012)
16. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22** (2010) 1345–1359
17. Jie, L., Tommasi, T., Caputo, B.: Multiclass transfer learning from unconstrained priors. In: ICCV. (2011)
18. Evgeniou, T., Pontil, M.: Regularized multi–task learning. In: KDD. (2004)
19. Kang, Z., Grauman, K., Sha, F.: Learning with whom to share in multi-task feature learning. In: ICML. (2011)
20. Wang, X., Zhang, C., Zhang, Z.: Boosted multi-task learning for face verification with applications to web image and video search. In: CVPR. (2009)
21. Vezhnevets, A., Buhmann, J.M.: Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In: CVPR. (2010)
22. Quadrianto, N., Smola, A.J., Caetano, T.S., Vishwanathan, S.V.N., Petterson, J.: Multitask learning without label correspondences. In: NIPS. (2010)
23. Parameswaran, S., Weinberger, K.: Large margin multi-task metric learning. In: NIPS. (2010)
24. Leen, G.: Context assisted information extraction. PhD thesis, University of the West of Scotland (2008)
25. Jia, Y., Salzmann, M., Darrell, T.: Factorized latent spaces with structured sparsity. In: NIPS. (2010)
26. Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the damage of dataset bias. In: ECCV. (2012)
27. Weinberger, K., Saul, L.: Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research **10** (2009) 207–244
28. Goldberger, J., Roweis, S.T., Hinton, G.E., Salakhutdinov, R.: Neighbourhood components analysis. In: NIPS. (2004)
29. Quadrianto, N., Lampert, C.H.: Learning multi-view neighborhood preserving projections. In: ICML. (2011)
30. Kleiner, A., Rahimi, A., Jordan, M.I.: Random conic pursuit for semidefinite programming. In: NIPS. (2010)
31. Lewis, A.S., Overton, M.L.: Nonsmooth optimization via quasi-newton methods. Math. Programming (to appear)
32. Microsoft Research Cambridge Object Recognition Image Database. http://research.microsoft.com/en-us/downloads/ (2005)
33. Krizhevsky, A.: Learning multiple layers of features from tiny images. Technical Report MSc thesis, University of Toronto, USA (2007)
34. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV **42** (2001) 145–175
35. Gibbons, J.: Nonparametric Statistical Inference. New York: Marcel Dekker (1985)