Landmarks-based Kernelized Subspace Alignment for Unsupervised Domain Adaptation

Rahaf Aljundi, Rémi Emonet, Damien Muselet and Marc Sebban
rahaf.aljundi@gmail.com
remi.emonet, damien.muselet, marc.sebban @univ-st-etienne.fr
CNRS, UMR 5516, Laboratoire Hubert Curien, F-42000, Saint-Étienne, France
Université de Saint-Étienne, Jean-Monnet, F-42000, Saint-Étienne, France

Abstract

Domain adaptation (DA) has gained a lot of success in the recent years in computer vision to deal with situations where the learning process has to transfer knowledge from a source to a target domain. In this paper, we introduce a novel unsupervised DA approach based on both subspace alignment and selection of landmarks similarly distributed between the two domains. Those landmarks are selected so as to reduce the discrepancy between the domains and then are used to non linearly project the data in the same space where an efficient subspace alignment (in closed-form) is performed. We carry out a large experimental comparison in visual domain adaptation showing that our new method outperforms the most recent unsupervised DA approaches.

1. Introduction

While the standard machine learning setting assumes that training and test data come from the same statistical distribution, it turns out that many real world applications, e.g., in Computer Vision or in Natural Language Processing, challenge this assumption. Indeed, the (source) data from which the classifier is supposed to be learned often differ from the (target) data the classifier will be deployed on (see Figure 1). To deal with such situations, the learning system has to take into account the distribution shift between the two domains in order to adapt well from the source to the target. In this framework, two main categories of *domain adaptation* (DA) methods are available. The first one, called semi-supervised DA, can gain access to some labeled examples from the target domain (as well as labeled source data) in order to perform the adaptation process (see, e.g., [6, 1, 18]). On the other hand, unsupervised DA assumes that no labeled information is available from the target domain. To deal with this more complex setting, DA methods (see [16] for a survey) usually assume that the distribution shift can be compensated either by a reweighting scheme applied on the source data (so as to better match the target distribution), or by a projection of both source and target data in a common (possibly latent) space. In this paper, we focus on this second category of DA methods which have the advantage of not assuming that source and target data originally lie in the same feature space.

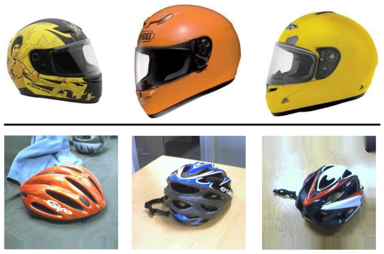


Figure 1. Example of distribution shift between images from two datasets. First row, some bike helmets from the Amazon subset and second row, bike helmets from the webcam subset. These 2 subsets are from the Office dataset.

It is worth noting that the most promising recent approaches rely on the assumption of the existence of a common embedded low dimensional manifold space that minimizes the divergence between the two distributions. In this context, subspace alignment-based DA methods have attracted a lot of interest. For example, in [10, 9], source and target data are projected onto intermediate (linear) subspaces that lie along the shortest geodesic path connecting the two original spaces. However, although effective, the search for such subspaces is computationally costly and subject to local perturbations. A recent work done by Fernando et al. [7] overcomes these drawbacks by optimizing a single linear mapping function that directly aligns the source and target subspaces. This new method has shown to be not only better than the state of the art but also computable in closed form. In this paper, we aim at going a

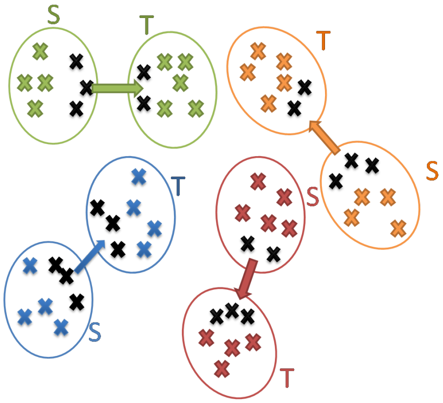


Figure 2. (Best shown in color) Domain adaptation problem that needs to adapt a source (S) domain to a target (T) domain w.r.t. four classes of examples (green, orange, blue and red). We can see that no linear transformation will be able to entirely bridge the source to the target. On the other hand, we can note that, for each class, some subsets of source and target examples seem to be similarly distributed (in black). The objective is to automatically discover those so-called landmarks that will be used to map both source and target data onto a common space using a kernel.

step further by improving this approach which faces two main limitations: First, it assumes that the shift between the two distributions can be corrected by a linear transformation. We claim that this is a strong assumption that can be challenged in many real world applications. Furthermore, we show in the experimental section that performing a simple kernel PCA [2] does not solve the problem. Second, it assumes that all source and target examples are necessary to proceed to the adaptation. However, it turns out that in most of the cases only a subset of source data are distributed similarly to the target domain and vice versa. In this paper, we deal with these two issues by:

- (i) Proposing a selection of landmarks extracted from both domains so as to reduce the discrepancy between the source and target distributions.

- (ii) Projecting the source and target data onto a shared space using a Gaussian kernel w.r.t. the selected landmarks. This allows us to capture the non linearity from the data in a simple way.

- (iii) Learning a linear mapping function to align the source and target subspaces. This is done by simply computing inner products between source and target eigenvectors.

As far as we know, this paper is the first attempt to select landmarks in an unsupervised way in order to reduce the discrepancy between the two domains. The intuition behind our method is summarized in Figure 2.

The rest of this paper is organized as follows. Section 2 is devoted to the presentation of the related work. In Section 3, we introduce our unsupervised DA method based on a landmark selection and a subspace alignment. Section 4 is dedicated to the experimental comparison performed on computer vision datasets. It highlights the interest of our approach which significantly outperforms the state of the art DA methods.

2. Related Work

Domain adaptation (DA) [16] is one of the possible settings of transfer learning [21]. In this paper, we focus on the most complicated case, called unsupervised DA, where labeled data are available only from the source domain. In this context, in order to reduce the shift between the source and target distributions, one strategy consists in reweighting each source data according to its similarity with the target domain [4, 5, 12, 23, 24]. Then, a classifier is learned (typically an SVM) on these weighted source instances. Another approach aims at projecting both source and target data in a common space which tends to move the two distributions closer to each other [20, 14, 3, 19]. This latter strategy has attracted a lot of interest during the past few years and gave rise to the development of *subspace alignment*-based DA methods. In [10], the authors learn a sequence of intermediate subspaces along the geodesic path connecting the source and target subspaces (see Figure 3). A similar work done by Gong et al. [9] also leans toward the same idea by proposing a geodesic flow kernel that tries to model incremental changes between the source and target domains. However, it is clear that computing such a large number of subspaces is computationally expensive.

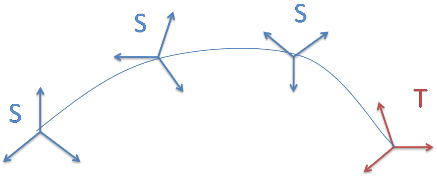


Figure 3. Intuition behind the idea of learning intermediate subspaces along the geodesic path connecting the source and target subspaces.

Note that recent approaches have tried to combine both source instance reweighting and distribution projection in a common space [15, 17, 11]. The very recent paper of Long et al. [15] is the most noticeable among them since their Transfer Joint Matching (TJM) algorithm provides the current state-of-the-art results. TJM aims at learning a new space in which the distance between the empirical expectations of source and target data is minimized while putting less importance to the source instances that are irrelevant to classify the target data. For this purpose, the authors make use of a kernel mapping of all the examples allowing by this way a non-linear transform between the two domains. It is worth noticing that, despite its efficiency, this approach suffers from several drawbacks. First, at testing time, the kernel has to be computed between each new instance and all the source and target examples, that can be very costly. Second, since the idea is to minimize the distribution difference while reweighting the source instances, the objective function is very complex to optimize and the authors resort to an alternating optimization strategy by iteratively updating one variable while the other ones are fixed. Third, because of this already complex optimization process, only the means of the source and target distributions are brought closer without accounting their standard deviations. We claim that the dispersions from the means can vary a lot between domains and should be adapted as well. This paper deals with all these issues: (i) it defines via a (cheap) greedy approach the landmarks that reduce the discrepancy between the two distributions; (ii) the mean and variance of the distribution of each landmark are taken into account in the adaptation process; (iii) the subspace alignment is performed in a closed form that only requires inner products between source and target eigenvectors (as suggested in [7]).

Since our method is directly inspired from [7], let us go into details of this one-step subspace alignment method (denoted by **SA** in the rest of this paper). Let us consider a labeled source dataset S and an unlabeled target dataset T lying in a D dimensional space. **SA** builds the source and target subspaces, respectively denoted by X_S and X_T, by extracting d eigenvectors (corresponding to the d largest eigenvalues, $d \leq D$) from both domains thanks to two independent PCA. Since X_S and X_T are orthonormal, the authors show that the optimal matrix M that transforms the source subspace into the target one is $M = X'_S X_T$. Note that since X_S and X_T are generated from the first d eigenvectors, they are intrinsically regularized. This explains why **SA** is extremely fast. Furthermore, the authors show that the unique hyperparameter d can be tuned by resorting to theoretical results inspired by concentration inequalities on eigenvectors.

As pointed out in the introduction, although very efficient, **SA** suffers from two main drawbacks. First, it allows only linear transforms between the source and target domains whereas most of the real transforms are non-linear (see Fig. 2). Second, forcing the algorithm to adapt all the source data to the target domain is a strong constraint while in most cases only a subset of source and target examples are similarly distributed.

In this paper, we cope with these two issues while preserving the theoretical guarantees and speed properties of **SA**. First, we automatically detect the subset of source and target examples that allow us to bring closer the two distributions. Then, we kernelize the problem w.r.t. those selected landmarks before performing the subspace alignment in closed-form. Using landmarks with a local kernel allows us to annihilate the impact of the points that are far from all landmarks.

Finally, note that the use of landmarks in domain adaptation has been recently used in [8] in a semi-supervised DA algorithm. Beyond the fact that we are working in this paper in a totally unsupervised setting, we select our landmarks from both sources while [8] only picks them from the source domain.

3. Proposed Approach for Domain Adaptation

In this section, we provide a detailed description of our landmark-based subspace alignment method for domain adaption. Our method is fully unsupervised in the sense that no labels are required to perform the domain adaptation. Labeled points from the source domain are only used afterwards to learn a classifier.

3.1. Task and Overall Approach

Source S and target T points are supposed to be respectively drawn from a source distribution D_S and a target distribution D_T. Domain adaptation supposes that the source and target distributions are not identical but share some similarities that make it possible to adapt to the target domain what has been learned on the source domain. Said differently, if we have a set of labels L_S for the source examples (all of them or only some of them), they can be used to learn a classifier that is suitable for the target domain.

Our approach combines two simple ideas: First, it projects both source and target examples in a common subspace w.r.t. some well selected landmarks. Then, it performs a subspace alignment between the two domains. After selecting landmarks among $S \cup T$, all points in S and T are projected using a Gaussian kernel on the selected landmarks, leading to new representations K_S and K_T for the source and target points. The new representation is then used to compute a mapping using a subspace alignment approach.

Compared to [7], our two-step approach remains fast and easy to implement while improving the accuracy by capturing non-linearity. The rest of this section details each step involved in our approach from the multiscale landmark selection process to the projected subspace alignment and the classification. The complete pseudo code is given in Algorithm 2 that shows the successive steps of our approach, also illustrated in Fig. 4.

3.2. Multiscale Landmark Selection

The first step of our method is to select some points as landmarks. Intuitively, a good set of landmarks is a

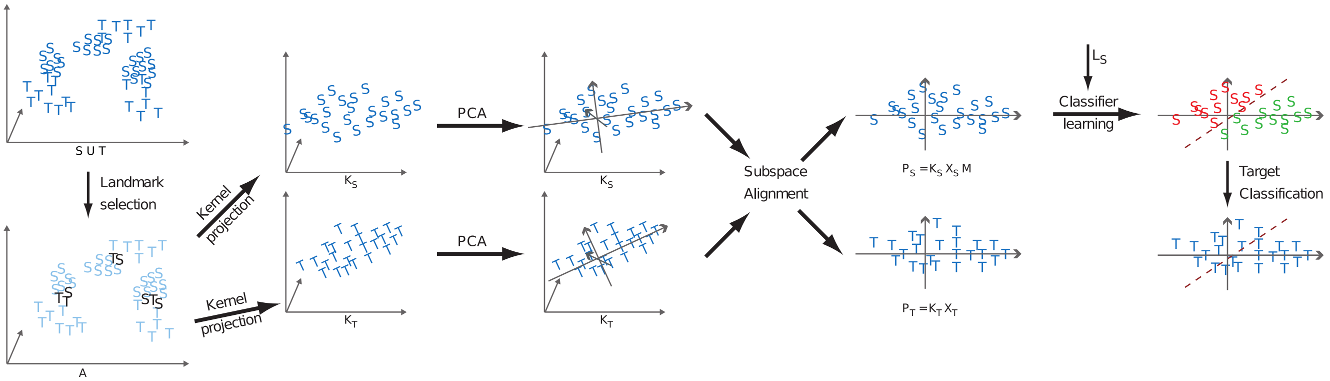


Figure 4. Overall workflow of our approach: First, landmarks are selected from $S \cup T$ so as to maximize the overlapping between the source and target distributions; Second, we get two new representations K_S and K_T for the source and target points using on Gaussian kernel on the selected landmarks; Then, two independent PCA are carried out before performing the subspace alignment w.r.t the d first eigenvectors; Finally, a classifier is learned (typically an SVM) from the labeled source data and deployed on the target domain.

set of points which can be used to project the source and target data onto a shared space in which their distributions are more similar. Our method chooses the landmarks among $S \cup T$ and does not use any label. The final output of the landmark selection procedure is a set of landmarks $A = \{\alpha_1, \alpha_2, ...\}$, $A \subset S \cup T$. To avoid any computationally expensive iterative optimization method, we propose a direct method of asserting whether a point should be kept as a landmark or not.

Landmark selection overview Our landmark selection approach actually considers each point c from $S \cup T$ as a candidate landmark. Each candidate is considered independently from the others: a quality measure is computed and if this measure is above a threshold, the candidate is kept as a landmark.

To assess the quality of the candidate c, its similarity with each point $p \in S \cup T$ is first computed, using a Gaussian kernel with standard deviation s:

$$K(c,p) = \exp\left(\frac{-\left\|c-p\right\|^2}{2s^2}\right) \tag{1}$$

The quality measure for the candidate c is computed as an overlap between the distribution of the $K(c,.)$-values for the source points, and the one for the target points. As a summary, a landmark is considered as a good one if the distribution of the source and the one of the target points are similar after projection using the kernel.

Multiscale analysis The value of the kernel radius s from Eq. 1 is important as it defines the size of the neighborhood that the candidate landmark considers. Choosing the right s for a given landmark will allow us to capture the local phenomena at the proper scale and to better align the source and target distributions. Extreme values for s must be avoided as they lead to a perfect match between the distributions of source and target points: all values of $K(c,.)$ become 0 (when s is close to 0) or 1 (when s is very big).

When calculating the quality of a candidate landmark, we actually perform a multiscale analysis: we select the best scale s to capture the local properties of the data, at the same time avoiding extreme values for s. To do so, we compute the distribution of euclidean distances between all pairs of points and try every percentile value of this distribution. With this percentile-based approach, we consider a range of values for the scale s which are all plausible. For each considered scale s, we compute an overlap measure between the source and target distributions, keeping the greatest one as the quality measure for the candidate landmark.

Distribution overlap criteria For a candidate landmark c and a scale s, we want to compute a degree of overlap between two sets of $K(c,.)$-values: the one for the source points KV_S and the one for the target points KV_T. To lower the computational cost, the two distributions are approximated as normal distributions and thus summarized by their means and standard deviations $\mu_S, \sigma_S, \mu_T, \sigma_T$. To be able to use a fixed threshold and give so semantics to it, we use a normalized overlap measure, explained below and computed as follows:

$$overlap(\mu_S, \sigma_S; \mu_T, \sigma_T) = \frac{\mathcal{N}(\mu_S - \mu_T \mid 0, \sigma^2_{sum})}{\mathcal{N}(0 \mid 0, \sigma^2_{sum})} \tag{2}$$

Where $\sigma^2_{sum} = \sigma^2_S + \sigma^2_T$ and $\mathcal{N}(. \mid 0, \sigma^2_{sum})$ is the centered 1D normal distribution.

To compute the overlap of two probability densities f and g, we can integrate their product: $\int f(x)g(x)dx$. Intuitively, the more overlap there is between the densities, the bigger their product. This overlap measure follows similar principles as the Bhattacharyya coefficient. By analytically integrating the product of two normal distributions, we obtain the numerator of Eq. 2:

$$\int \mathcal{N}(x \mid \mu_S, \sigma_S^2)\mathcal{N}(x \mid \mu_T, \sigma_T^2)dx = \mathcal{N}(\mu_S - \mu_T \mid 0, \sigma_{sum}^2) \tag{3}$$

The denominator in Eq. 2 corresponds to the maximum value of the numerator for a given σ_{sum} (obtained when $\mu_S = \mu_T$). The denominator acts as a normalization factor: it sets the overlap to 1 when the distributions are perfectly matching and gives an easier interpretation, helping in the choice of the threshold th.

Summary Algorithm 1 sums up the landmark selection process. Each point from $S \cup T$ is considered as a candidate landmark. For each candidate, we consider multiple scales s. If, for any of these scale, the overlap (Eq. 2) between the source and target distributions of $K(c, .)$-values is greater than th, the candidate is promoted as a landmark.

Algorithm 1 MLS: Multiscale landmarks selection, implementing *choose_landmarks* in Algorithm 2.

```
Require: Source data $S$, Target data $T$, Threshold $th$.
Ensure: $A$ contains the selected landmarks.
    $A \leftarrow \{\}$
    $distances \leftarrow \{\|a - b\|, (a, b) \in (S \cup T)^2\}$
    for $c$ in $S \cup T$ do
        for $s$ in $percentiles(distances)$ do
            $KV_S \leftarrow \{exp(-\|c - p\|^2 / 2s^2), p \in S\}$
            $KV_T \leftarrow \{exp(-\|c - p\|^2 / 2s^2), p \in T\}$
            if $overlap(KV_S, KV_T) > th$ then
                $A = A \cup \{c\}$
            end if
        end for
    end for
```

3.3. Kernel Projection and Subspace Alignment

Once the set of landmarks A has been selected, we use a Gaussian kernel to achieve a non-linear mapping of all the points into a common space defined by these landmarks. The subspaces from the source and the target domains are then aligned using a linear transformation.

Projection on the selected landmarks Each point p from $S \cup T$ is projected onto each landmark $\alpha_j \in A$ using a Gaussian kernel with standard deviation σ:

$$K(p, \alpha_j) = \exp\left(\frac{-\|p - \alpha_j\|^2}{2\sigma^2}\right) \tag{4}$$

Overall, the points from both S and T are projected in a common space that has as many dimensions as there are landmarks. Following other non-linear methods we set σ to the median distance between any pair of points drawn randomly from $S \cup T$. The value of σ could also be selected using some cross-validation procedure. After projection, we finally obtain new representations for the source and target, respectively denoted by K_S and K_T.

Subspace alignment Using the non-linearly projected set of points K_S and K_T, we follow a subspace alignment approach. PCA is applied on each domain separately to extract the d eigenvectors having the largest eigenvalues. Following the theoretical part of [7], we are able to determine the optimal value of d. Indeed, the authors in [7] derive a consistency theorem based on a standard concentration inequality on eigenvectors. This theorem allows them to get a bound on the deviation between 2 successive eigenvalues. In this paper, we make use of this bound to efficiently tune d, the number of dimensions in PCA. The d eigenvectors for the source and the target domains are respectively denoted X_S and X_T. The points from each domain can be projected on their respective subspace as $K_S X_S$ and $K_T X_T$.

The goal of subspace alignment is to find a linear transformation M that best maps the source eigenvectors onto the target eigenvectors. In the end, we want to find M that minimizes the sum of the euclidean distances between the M-transformed source eigenvectors and the target eigenvectors. This minimization is equivalent to minimizing the following Frobenius norm:

$$F(M) = \| X_S M - X_T \|_F^2 . \tag{5}$$

Equation 5 has a closed form solution that can be implemented efficiently as a multiplication of the eigenvectors:

$$M = X_S' X_T. \tag{6}$$

The alignment transformation M maps points from the source eigenspace to the target eigenspace. The transformation M can be used to bring the projected source points $K_S X_S$ into the same eigenspace as the projected target points $K_T X_T$, by computing $K_S X_S M$.

The complete pseudo-code of our algorithm, called **LSSA** (for **L**andmarks **S**election-based **S**ubspace **A**lignment), is described in Algorithm 2.

4. Experiments

The objective of this experimental section is twofold. First, we aim at studying the behavior of our landmark selection method, taken alone, to deal with unsupervised visual domain adaptation in comparison with other landmark selection approaches. Second, we seek to show that jointly used with a subspace alignment process, our landmark selection method yields to a significant enhancement of the adaptation process and allows us to outperform the state of

Table 1. Comparison (in terms of accuracy) of 5 landmark selection methods on 12 unsupervised DA subproblems. C: Caltech, A: Amazon, W: Webcam, D: Dslr. **RD**: Random Selection; **All**: all the source and target examples are used; σ-**LS**: our selection method with a fixed σ; **CDL**: Connecting Dots with Landmarks; **MLS**: our approach. In red, one reports the best method.

Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D → W	D → C	D → A	Avg
RD	40.3	38.8	42.3	41.2	40.6	47.5	84.0	32.9	28.4	81.8	36.8	32.3	45.6
All	41.0	39.4	44.7	41.4	41.6	49.6	85.3	33.0	29.2	82.7	38.6	31.3	46.5
σ-**LS**	39.3	37.5	43.8	42.7	31.5	52.4	80.3	32.6	29.5	82.0	38.6	31.2	45.1
CDL	38.3	38.8	43.9	45.8	45.4	51.7	77.7	35.3	30.9	72.5	33.9	33.3	45.6
MLS	41.1	39.5	45.0	45.2	44.1	53.6	84.7	35.9	31.6	82.4	39.2	34.5	48.1

Algorithm 2 LSSA: **L**andmarks **S**election-based **S**ubspace **A**lignment and classification.

```
Require: Source data $S$, Target data $T$, Source labels $L_S$,
    Threshold $th$, Subspace dimension $d$.
Ensure: $L_T$ are the predicted labels for the points in T.
    $A \leftarrow choose\_landmarks(S, T, th)$
    $\sigma \leftarrow median\_distance(S \cup T)$
    $K_S \leftarrow project\_using\_kernel(S, A, \sigma)$
    $K_T \leftarrow project\_using\_kernel(T, A, \sigma)$
    $X_S \leftarrow PCA(K_S, d)$
    $X_T \leftarrow PCA(K_T, d)$
    $M \leftarrow X_S' X_T$
    $P_S \leftarrow K_S X_S M$
    $P_T \leftarrow K_T X_T$
    $classifier \leftarrow learn\_classifier(P_S, L_S)$
    $L_T \leftarrow classifier(P_T)$
```

the art, including the two best recent methods presented in [7, 15].

4.1. Datasets and Experimental Setup

We run our experiments on standard datasets for visual domain adaptation. We use the Office dataset [22] that contains images acquired with webcams (denoted by W), images taken with digital SLR camera (denoted by D) and Amazon images (denoted by A). In addition, we use Caltech10 [9], denoted by C. Each dataset provides different images from 10 classes[1]. Therefore, we can generate 12 domain adaptation subproblems from the four datasets (A, C, D, W), each time one dataset playing the role of the source S while another one is considered as the target T. We denote a DA problem by the notation $S \rightarrow T$. The objective is to learn an SVM classifier (with a linear kernel using the SVM^{light} implementation) from the labeled source S and to deploy it on the target T. We use the same image representation as that provided by [9] for Office and Caltech10 datasets (SURF features encoded with a visual dictionary of 800 words). We follow the standard protocol of [9, 10, 7, 13, 22] for generating the source and target samples.

As indicated before, we perform the following two series of experiments.

Comparison of landmark selection methods To achieve this task, we compare our method (**MLS** in Table 1) with three baselines:

- A random selection (**RD**). For this purpose, we randomly select 300 landmarks (150 for each domain) and repeat the task five times to get a behavior on average.
- No landmark selection (**All**). We use all the source and target examples as landmarks.
- Our method without the multiscale strategy (σ-**LS**). The same standard deviation σ is used for all candidates. σ is set (to the advantage of this baseline) to the best standard deviation allowing on average the maximum overlapping between the two distributions.

Note that the first two baselines do not perform any adaptation because they do not aim at specifically moving the distributions close to each other (the model is simply learned from the source data and deployed on the target domain). For **MLS** and σ-**LS**, we fix the overlapping rate to 0.3 to select a landmark. Because of the normalization, a threshold of 0.3 corresponds to an overlapping rate of 30%. We also compare **MLS** with another landmark selection method, called *Connecting Dots with Landmarks* (**CDL**), recently presented in [8] in a semi-supervised DA setting.

Comparison with the state of the art unsupervised DA approaches The second series of experiments is devoted to the evaluation of our landmark selection method used in an **unsupervised subspace alignment DA** setting. We compare our approach (**LSSA**), as presented in Algorithm 2, with three state of the art methods:

- Geodesic Flow Kernel (**GFK**) [9], where a sequence of intermediate subspaces is learned along the geodesic path connecting the source and target domains.

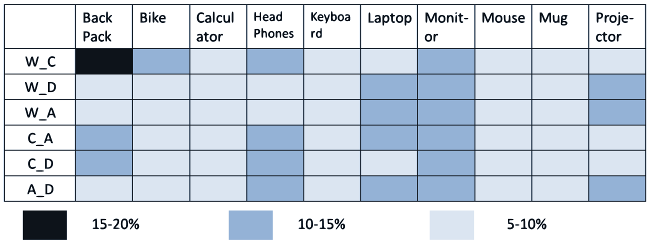
[1] BackPack, Bike, Calculator, Headphone, Keyboard, Laptop, Monitor, Mouse, Mug, Projector.

- The one step subspace alignment (**SA**) proposed by [7] which learns a linear transformation between the two subspaces.
- The Transfer Joint Matching (**TJM**) presented in [15], which is a recent work based on both feature matching and instance weighting.

In addition, we perform experiments with two baselines: the first one does not perform any adaptation (**NA**); the second one performs two independent KPCAs on the source and target domains and then learns the linear transformation using the algorithm **SA** [7] (denoted by **KPCA+SA**).

4.2. Analysis of the Results

Comparison of landmark selection methods From the results reported in Table 1, we can make the following comments.
First, we can note that our method significantly (using a Student paired-t test) outperforms on average the other approaches (with an average accuracy of 48.1%). Among 12 DA tasks, **MLS** has the best behavior in terms of accuracy in 8 subproblems. Second, for two subproblems (W→D and D→W), **All** is better that means that keeping all the source and target examples in these two (symmetric) cases is better than trying to select landmarks. It is worth noting that the two corresponding subproblems are the simplest ones (with the highest accuracy) justifying the interest to keep all the data. Moreover, we can note that in 10 subproblems out of 12, our approach outperforms **CDL** which is specifically dedicated to select landmarks in a semi-supervised DA setting. Lastly, the single scale approach (using a fixed σ) does not perfom well: this illustrates how important it is to select the best radius of action for every landmark in **MLS**.
Figure 5 gives the distribution of the landmarks selected by **MLS** for each DA subproblem, showing that even without class information, our approach makes a balanced selection among the classes.



	Back Pack	Bike	Calculator	Head Phones	Keyboard	Laptop	Monitor	Mouse	Mug	Projector
W_C	■	▨	□	▨	□	□	▨	□	□	□
W_D	□	□	□	□	□	▨	▨	□	□	▨
W_A	□	□	□	□	□	▨	▨	□	□	▨
C_A	▨	□	□	▨	□	▨	▨	□	□	□
C_D	▨	□	□	▨	□	□	▨	□	□	□
A_D	□	□	□	▨	□	▨	▨	□	□	▨

■ 15-20% ▨ 10-15% □ 5-10%

Figure 5. Distribution of the landmarks among the ten classes for each DA subproblem.

Comparison with the state of the art unsupervised DA approaches Table 2 reports the results of the experimental comparison between the state of the art unsupervised subspace alignment DA methods. It is worth noticing that our method **LSSA** outperforms the other approaches in 7 out of 12 DA subproblems, while **TJM** is better for the remaining 5 subproblems. However, on average, **LSSA** significantly outperforms **TJM** (52.6% versus 50.5%). Moreover, as pointed out by the authors, the time complexity of **TJM** is larger than the other approaches because it requires to solve a non trivial optimization problem, while our approach which involves a greedy strategy (for the landmark selection) as well as a closed solution (for the subspace alignment) is more efficient. We claim that the difference of accuracy between **TJM** and **LSSA** comes from the fact that the former uses a weighting scheme that mainly aims at moving closer the means of the two domains, while the latter takes into account, via the Gaussian hypothesis, not only the mean but also the standard deviation of the statistical distribution of the landmarks. From Table 2, we can also notice that **LSSA** significantly outperforms **SA**, meaning that capturing non linearity in **LSSA** is a big improvement over plain **SA**. However, the way to consider the non linearity is also key. Indeed, as shown by **KPCA+SA**, performing two independent KPCA before the subspace alignment leads to the worst behavior.

5. Conclusion

Subspace alignment-based DA methods have recently attracted a lot of interest. In this framework, the most accurate approach assumes that the shift between the source and target distributions can be corrected by a linear function. However, we experimentally show that this assumption is challenged in most of the real world applications. Thus, we argue that a non linear mapping function should be optimize to align the source and target subspaces. Furthermore, there is no reason that would justify to constrain a DA algorithm to adapt all the training source data to the target domain. In this paper, we have combined both ideas in a new unsupervised DA algorithm (**LSSA**) which first selects landmarks that allow us to create, by a non linear projection, a common space where the two domains are closer to each other and then performs a subspace alignment. **LSSA** is simple, fast and outperforms the state of the art on a visual domain adaptation task. As a future work, we plan to resort to an optimization process that jointly selects the landmarks in order to avoid redundancy between them as it might appear with the current greedy and independent selection.

Acknowledgement This work is supported by the ANR SOLSTICE project (ANR-13-BS02-01).

References

[1] A. Bergamo and L. Torresani. Exploiting weakly-labeled web images to improve object classification: a domain adap-

Table 2. Comparison (in terms of accuracy) of unsupervised DA methods. C: Caltech, A: Amazon, W: Webcam, D: Dslr. **NA**: No Adaptation; **KPCA+SA**: two independent KPCA are performed on the source and target data, then a subspace alignment is applied; **GFK**: Geodesic Flow Kernel; **SA**: one step Subspace Alignment; **TJM**: Joint Matching Transfer; **LSSA**: our approach.

Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D → W	D → C	D → A	Avg
NA	31.5	40.7	45.4	38.2	30.2	50.1	80.2	32.4	31.2	67.8	28.3	30.8	42.2
KPCA+SA	10.1	5.1	7.7	7.6	10.5	10.4	7.6	10.4	11.8	7.2	8.5	7.5	8,7
GFK	38.6	35.7	40.1	44.6	39.0	54.1	81.2	36.6	28.9	80.3	39.2	33.1	45.9
SA	40.7	46.4	41.6	49.0	42.7	52.7	78.9	39.4	34.7	83.4	44.8	38.0	49.3
TJM	42.0	45.8	45.7	49.0	48.8	58.6	83.4	40.8	34.8	82.0	39.6	35.1	50.5
LSSA	42.4	47.2	44.8	54.1	48.1	58.4	87.2	39.4	34.7	87.1	45.7	38.1	52.6

tation approach. In *NIPS*, volume 1, page 4, 2010. 1

[2] R. Chatpatanasiri, T. Korsrilabutr, P. Tangchanachaianan, and B. Kijsirikul. A new kernelization framework for mahalanobis distance learning algorithms. *Neurocomputing*, pages 1570–1579, 2010. 2

[3] B. Chen, W. Lam, I. Tsang, and T.-L. Wong. Extracting discriminative concepts for domain adaptation in text mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 179–188. ACM, 2009. 2

[4] M. Chen, K. Q. Weinberger, and J. Blitzer. Co-training for domain adaptation. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2456–2464. Curran Associates, Inc., 2011. 2

[5] W.-S. Chu, F. D. la Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3515–3522, 2013. 2

[6] H. Daumé III, A. Kumar, and A. Saha. Co-regularization based semi-supervised domain adaptation. In *NIPS*, volume 10, pages 478–486, 2010. 1

[7] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*, 2013. 1, 3, 5, 6, 7

[8] B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *Proceedings of The 30th International Conference on Machine Learning*, pages 222–230, 2013. 3, 6

[9] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2066–2073. IEEE, 2012. 1, 2, 6

[10] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, pages 999–1006. IEEE, 2011. 1, 2, 6

[11] Q. Gu, Z. Li, and J. Han. Joint feature selection and subspace learning. In *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 1294–1299, 2011. 2

[12] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. *Advances in neural information processing systems*, 19:601, 2007. 2

[13] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1785–1792. IEEE, 2011. 6

[14] M. Long, G. Ding, J. Wang, J. Sun, Y. Guo, and P. Yu. Transfer sparse coding for robust image representation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 407–414, June 2013. 2

[15] M. Long, J. Wang, G. Ding, J. Sun, and P. Yu. Transfer joint matching for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1410–1417, June 2014. 2, 6, 7

[16] A. Margolis. A literature review of domain adaptation with unlabeled data. *Technical Report*, 2011. 1, 2

[17] M. Masaeli, J. G. Dy, and G. M. Fung. From transformation-based dimensionality reduction to feature selection. In J. Frnkranz and T. Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 751–758. Omnipress, 2010. 2

[18] R. Mehrotra, R. Agrawal, and S. A. Haider. Dictionary based sparse representation for domain adaptation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2395–2398, New York, NY, USA, 2012. ACM. 1

[19] S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *AAAI*, volume 8, pages 677–682, 2008. 2

[20] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *Neural Networks, IEEE Transactions on*, 22(2):199–210, 2011. 2

[21] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, October 2010. 2

[22] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, pages 213–226. Springer, 2010. 6

[23] M. Sugiyama, S. Nakajima, H. Kashima, P. Von Buenau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *NIPS*, volume 7, pages 1433–1440, 2007. 2

[24] Y. Tsuboi, H. Kashima, S. Hido, S. Bickel, and M. Sugiyama. Direct density ratio estimation for large-scale covariate shift adaptation. *Information and Media Technologies*, 4(2):529–546, 2009. 2