ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Transfer Learning by Sharing Support Vectors*

Vitaly Ablavsky (vitaly.ablavsky@epfl.ch)

Carlos Becker (carlos.becker@epfl.ch)

Pascal Fua (pascal.fua@epfl.ch)

School of Computer and Communication Sciences
Swiss Federal Institute of Technology, Lausanne (EPFL)

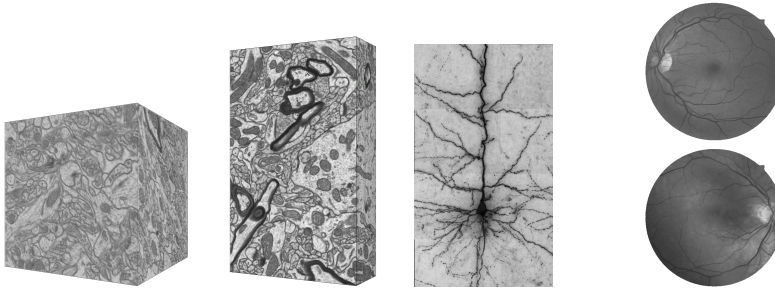
Technical Report

September 18, 2012

* This work was supported in part by the ERC MicroNano project.

Abstract. Machine Learning techniques play an increasingly vital role in the analysis of Biomedical imagery, as in all other areas of Computer Vision. However, in this specific context, they suffer from the fact that experimental conditions and protocols change often and that acquiring sufficient amounts of new training data after each image acquisition is impractical.
In this paper, we propose an effective method to train a non-linear SVM using a very small amount of new data by leveraging data obtained under different conditions. Unlike earlier approaches, ours takes full advantage of the kernelized SVM formulation, does not depend on a loss function that is sensitive to outliers, and yields a quadratic optimization problem. We demonstrate its effectiveness for the purpose of classifying pixels in electron microscope image stacks and delineating linear structures in optical microscopy and retinal scans. Our method outperforms two state-of-the-art transfer-learning approaches in terms of accuracy and computational complexity.

1 Introduction



(a) CA1 Hippocampus (b) Striatum (c) Dendritic tree (d) Retina blood vessels

Fig. 1: Obtaining high-quality annotation for all the biomedical image data that is rapidly becoming available is prohibitively expensive. We propose a transfer-learning approach to train a classifier for a new domain with few annotations. With our approach we can transfer a mitochondria classifier from (a) CA1 Hippocampus to (b) striatum and from (c) dendritic tree to (d) retinal blood vessels.

Statistical Machine Learning methods have become dominant for detection and segmentation purposes in all domains for which sufficient amounts of training data can be obtained. However, these methods require retraining with labelled data when imaging conditions change. Unfortunately, in a field such as biomedical imaging, acquisition conditions tend to vary frequently due to changes in the protocol or instrument settings. At the same time, annotating data such as the 2D and 3D images in Fig. 1 requires painstaking effort, creating a bottleneck for effectively employing learning methods.

In this paper, we address this issue by jointly training two SVMs whose support vectors are constrained to remain close from each other, one for a *reference dataset* for which abundant training data has been obtained and the other for an

input dataset for which training data is in short supply. We obtain classification results on the input dataset, which, in some cases, are indistinguishable from those we would have obtained by fully retraining using much larger training sets. This means that each time new images are acquired an operator only has to spend a few minutes, as opposed to days or months, acquiring a few positive and negative examples to retrain the system, which is acceptable in such an operational setting.

Although our goal of sharing information between classifiers can be viewed as an instance of Transfer Learning, none of the existing techniques are fully satisfactory for our purposes. Domain adaptation approaches, such as the recently-proposed [1], are not designed to exploit the limited number of labelled examples that are available in the input dataset. Recent approaches to new category learning [2, 3] exploit closed-form generalization estimates of quadratic-loss SVMs, but at the cost of using a loss function that is sensitive to outliers. The recent category-learning approach of [4] considers structured object models, but only for linear kernels. Our approach was inspired by that of [5], which also involves learning multiple SVMs whose support vectors are constrained to remain close. In this work, we fully realize the benefit of the ideas presented in [5] by including per-task bias terms. The method of [6] achieves many of the same goals but is far more computationally demanding, which makes its use problematic for biomedical applications.

Our contribution is an easy to implement approach to Transfer Learning that requires very little labeled data for effective retraining and preserves the learning potential of kernelized SVMs. We demonstrate its effectiveness on challenging real-world problems: the detection of mitochondria, dendrites and blood vessels in microscopy images such as those depicted by Fig. 1. Our approach outperforms the state-of-the-art approach of [2] and does at least as well or better than the method of [6] at a fraction of the computational cost.

2 Related work

In this section we review related work on transfer learning. For more details, a comprehensive survey may be found in [7].

Domain transformation. If the input-domain distribution could be transformed to match the reference-domain distribution, then a classifier trained on the reference-domain would attain comparable accuracy on the input domain. Unfortunately, in medical imaging scenarios, the underlying factors contributing to drastic change in the domains are not always easily quantified. Applying kernel-mean-matching to the two domains may yield sub-optimal results, as was reported in [8]. Although one could attempt to learn a transformation of the metric between the two domains, such approaches have been only demonstrated for nearest-neighbor classifiers [9], which tend to be less powerful than SVM's.

Unsupervised transfer learning. For the sake of completeness, we mention approaches that attempt transfer learning without the benefit of input-domain annotation. An SVM-based approach of [10], iteratively estimated the

Table 1: State-of-the-art supervised transfer learning approaches. Among these, only the methods of [2, 6] and ours rely on convex objective functions, and allows kernelization. We will show in the result section that we outperform both. Notation: an n-th example from task t is specified as $(\mathbf{x}_{t,n}, y_{t,n})$. A decision function f_t is parameterized by $\mathbf{w}_t$ and may include an offset (bias)

Supervised approaches to transfer learning	Optimization		Decision function	
	Convex?	Objective	Bias?	Kernels?
Jebara 2011 [6]	convex, but non-quadratic	max-entropy discrimintation w.r.t. $p(y_{t,n}\|\mathbf{x}_{t,n}, \mathbf{w}, \mathbf{w}_t)$	Y	Y
Tommasi et al. 2010 [2]	quadratic	$\min \sum \left(1 - y_{t,n} f_t(\mathbf{x}_{t,n})\right)^2$	Y	Y
Aytar et al. 2011 [4]	non-convex	$\min \sum \left[1 - y_{t,n} f_t(\mathbf{x}_{t,n})\right]_+$ + deformation	Y	N
Evgeniou et al. 2004 [5]	quadratic	$\min \sum \left[1 - y_{t,n} f_t(\mathbf{x}_{t,n})\right]_+$	N	Y
Proposed approach			Y	Y

labels of the input domain, while gradually erasing the labels of the reference domain; a heuristic was introduced to automatically detect when the procedure failed. In the nearest-neighbor classification approach of [1], a sequence of intermediate problems was constructed to gradually adapt the reference domain to the input domain.

However, as with all unsupervised approaches, failure modes may be difficult to understand by a non-expert. Therefore, in the medical domain, it is more practical to ask the end-user to supply few annotations for the input domain, rather than spending time figuring out why the unsupervised algorithm is underperforming. Furthermore, classification approaches that ignore labelled training data in the input domain tend to be outperformed by the supervised ones.

Supervised transfer learning. The benefit of learning a neural network for multiple related tasks simultaneously was demonstrated in [**?**]. During the intervening years, multi-task learning and transfer learning have been developed for classifiers that are relevant to our field of application.

An empirical analysis of SVM-based transfer learning for genome sequencing was presented in [8]. However, the formulation that was found to perform best, dual-task learning ($SVM_{S,T}$), was deemed suboptimal in [4], since the regularization compromised margin maximization. A fix was proposed in [4], but only for the linear SVM's.

Learning a classifier by leveraging existing classifiers was proposed in [3] and [2]. However, in order to determine the relevance of the prior classifiers, their formulation relied on the Least-Squares (LS) SVM, which we will show to be less effective for our purposes than our hinge-loss SVM formulation.

Relevant multi-task approaches include [6] that took the form of maximum-entropy discrimination. However, the resulting optimization problem was no longer quadratic. The approach of [5] involved learning multiple linear decision functions with a regularization penalty encouraging all decision functions to be similar. However, for kernel-based SVMs, the formulation in [5] was developed using feature-space analogies rather than formally derived. Furthermore, neither

in [5] nor in a follow-up one [11, 12] is a bias term included, although such a term is often crucial [13].

Discussion. As Table 1 indicates, our extension of [5] offers generality and computational advantages, in contrast to recent prior work. In particular, our formulation compares favorably in terms of computational complexity to [6], since the optimization problem remains quadratic, it allows kernels, while [4] does not, and, unlike [2], it optimizes the relevant objective. The benefits of our approach become evident when it is applied to our medical-imaging datasets, where domain differences are large, label noise is high, and ratio of input-domain and reference-domain examples is very low.

3 Approach

Our approach is designed for cases where a recognition system achieves state-of-the-art accuracy on a data set for which it has been trained, but whose performance decreases on a related data set. Examples of our applications include transfer learning across different FIB-EM volumes, as shown in Fig. 1(a,b) and between different types of tubular structures, like those in Fig. 1(c,d).

Problem definition. Our goal is to learn a decision function for an *input task*, given labelled examples for this input task and for *reference task*; let $t \in \{1, 2\}$ denote the task label. The labelled examples for each task are specified as $\{(\mathbf{x}_{t,n}, y_{t,n})\}$; N_t is the number of examples in each task. In our applications the number of examples in the input task, N_1 is much less than N_2, the number of examples in the reference task.

Linear formulation. We consider support-vector-machine (SVM) classifiers, decision functions $y = f(\mathbf{x})$ of the form $y = \langle \mathbf{w}, \mathbf{x} \rangle + b$, where $\langle \cdot, \cdot \rangle$ denotes an inner product. There is one such decision function per task. Our goal is to learn f_t for the target task by leveraging training examples from all T tasks.

Since the tasks are related it is reasonable to assume that the decision functions f_t also are similar. We express this by writing $\mathbf{w}_t = \mathbf{w}_0 + \mathbf{v}_t$, where $\mathbf{w}_0$ is common and $\mathbf{v}_t$'s are task-specific. Unlike [5] our formulation includes per task bias terms $\{b_t\}$. As was mentioned earlier, a bias term is crucial for attaining the desired accuracy on some recognition tasks.

Our objective function J which we wish to minimize takes the form

$$J = C_t \sum_{t=1}^{2} \sum_{n=1}^{N_t} \xi_{t,n} + \lambda_2 \frac{1}{2} \|\mathbf{w}_0\|^2 + \frac{\lambda_1}{2} \frac{1}{2} \sum_{t=1}^{2} \|\mathbf{v}_t\|^2 \tag{1}$$

where ξ's are margin violation errors defined as $y_{t,n} f(x_{t,n}) - 1 + \xi_{t,n} \geq 0$, $\xi_{t,n} \geq 0$. The non-negative penalty terms C_t, λ_1, and λ_2 penalize the margin-violation errors and the large norms of the decision boundaries. Large λ_1 discourages deviations from the shared $\mathbf{w}_0$, and large λ_2 promotes diversity between $\mathbf{w}_t$'s. Although dividing by λ_1 would not change the objective function J, the current form of J could be viewed as more intuitive.

Following [14] we maximize the corresponding Lagrangian L, which is equal to

$$J - \sum_{t,n} \alpha_{t,n} \left[y_{t,n} \left(\langle \mathbf{x}_{t,n}, \mathbf{w}_0 + \mathbf{v}_t \rangle + b_t \right) + \xi_{t,n} - 1 \right] - \sum_{t,n} \beta_{t,n} \xi_{t,n} \tag{2}$$

and where α's and β's are non-negative Lagrange multipliers. Imposing the stationarity results in a set of constraints that includes

$$0 = \frac{\partial L}{\partial b_t} = -\sum_{n} \alpha_{t,n} y_{t,n}. \tag{3}$$

The interpretation of Eq. 3 is that realizing the benefit of each f_t having its own bias term, results in straightforward per-task constraints on the dual variable α's.

After all constraints resulting from stationarity of L are substituted into Eq. 2, it takes the form of

$$\begin{aligned} L = &\sum_{t,n} \alpha_{t,n} - \frac{1}{2\lambda_2} \sum_{t,n} \sum_{t',n'} y_{t,n} y_{t',n'} \alpha_{t,n} \alpha_{t',n'} \left\langle \mathbf{x}_{t,n}, \mathbf{x}_{t',n'} \right\rangle \\ &- \frac{2}{2\lambda_1} \sum_{t} \sum_{n,n'} y_{t,n} y_{t',n'} \alpha_{t,n} \alpha_{t',n'} \left\langle \mathbf{x}_{t,n}, \mathbf{x}_{t',n'} \right\rangle, \end{aligned} \tag{4}$$

subject to the constraints of Eq. 3. The set of α's and b_t's that maximize L, defines task-specific decision functions

$$f_t(\mathbf{x}) = \sum_{t'=1}^{2} \sum_{n=1}^{N_{t'}} y_{t',n} \alpha_{t',n} \langle \mathbf{x}, \mathbf{x}_{t',n} \rangle_{(t,t')} + b_t. \tag{5}$$

Kernel formulation. The optimization problem specified by Eq. 4 leads to a revised definition of the inner product. The new inner product $\langle \cdot, \cdot \rangle_{(t,t')}$ takes the form

$$\langle \mathbf{x}, \mathbf{x}' \rangle_{(t,t')} = \begin{cases} \left(\dfrac{1}{\lambda_2} + \dfrac{2}{\lambda_1} \right) \langle \mathbf{x}, \mathbf{x}' \rangle & \text{if } t = t' \\ \dfrac{1}{\lambda_2} \langle \mathbf{x}, \mathbf{x}' \rangle & \text{otherwise} \end{cases} \tag{6}$$

The above derivation holds when $\mathbf{x}$ is replaced by a feature map, and the inner product is computed implicitly in terms of the kernel $k(\mathbf{x}, \mathbf{x}')$. Therefore, our revised inner product yields a revised kernel $k(\mathbf{x}, \mathbf{x}')_{(t,t')}$.

Comparison to Related Approaches. Although our formulation is a textbook-style extension of [5], we are the first to fully realize the benefit of this approach (whose potential seems to have been overlooked by [8, 2, 4]). In the Supplementary material we show that the dual-task learning of [8], called $SVM_{S,T}$ is a degenerate case of our formulation.

Task-specific C_t's might be omitted in multi-task approaches that deal with balanced tasks, as in [11, 6]. However, in our applications, the number of examples in the input domain is much smaller than in the reference domain. Therefore, the C_t's must be normalized to give equal importance to both domains.

4 Implementation

To solve the quadratic optimization problem from the multi-task formulation we employ MOSEK [1] optimization tool. Model cross-validation is further sped up making use of MATLAB's parallel processing toolbox on a multi-core computer.

Parameter selection is an important part of tuning SVMs to handle a specific problem. In our case, it involves setting C, σ and λ. As is common practice [15, 16], we set the bandwidth σ of the RBF kernel to the average distance between the training samples. The other parameters are selected in a stratified fashion. First, we select an optimal C for the reference domain by cross-validation using a one-dimensional grid search with $C \in \{0.01, 0.1, 0.5, 1.0, 5, 10, 20, 50, 100\}$. We then set $C_1 = N_2/(N_1 + N_2)$ and $C_2 = N_1/(N_1 + N_2)C$. The remaining parameters (λ_1, λ_2) are set by fixing $\lambda_1 = 1$ and optimizing λ_2 via five-fold cross validation, with $\lambda_2 \in \{1, 2, 3, 5, 9, 16, 28, 48\}$. This scheme is used for all experiments described in Section 5.

5 Experiments

In this section, we first demonstrate the effectiveness of our algorithm on synthetic data. We then show that it behaves similarly on real biomedical data. We compare our algorithm with the approaches identified in the Related work section as being the most relevant: [5, 2, 6]. Following their experimental protocol, our baseline comparisons also include independent training separately for each domain, and the aggregation of the examples from both domains.

5.1 Synthetic Data

We validate our implementation on a synthetic dataset. In this dataset there are two binary classification tasks. For each task, two-dimensional features are sampled from a distribution where the two classes are separable using a second-degree polynomial kernel. The decision boundaries for the two tasks, which are hyperplanes in the kernel-induced feature space, are related by a linear transform. A small amount of label noise is introduced near the true decision boundaries. The number of examples in the reference domain is set to three times the number of examples in the input domain.

We apply our transfer-learning algorithm to this dataset with $\lambda_2 \in \{1.5, 10\}$. We plot the results in Fig. 2 where the (a),(c) correspond to the reference domain and (b),(d) to the input domain; true decision boundaries are shown in green, and the estimated decision boundaries are shown in black.

When $\lambda_2 = 10$ the reference domain has little impact on the learning in the input domain. Thus, while the decision boundary of the reference domain approaches the true decision boundary, the decision boundary of the input domain overfits to the few training examples. However, for $\lambda_2 = 1.5$ the regularization

[1] `http://www.mosek.com`

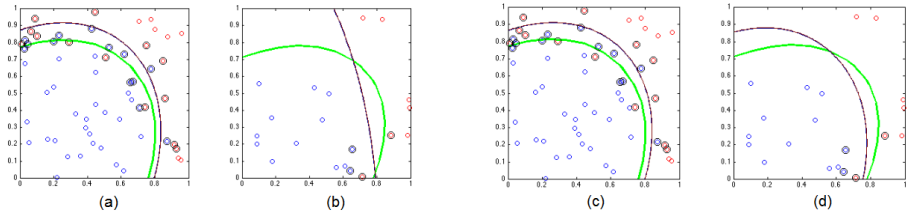


Fig. 2: Sharing of support vectors between the reference task (a),(c) and the input task (b),(d) benefits the classifier for the input domain. Without sufficient sharing (a, b), the estimated classifier in the input domain (shown in black) overfits compared to the ground truth (green). As sharing increases (c, d), the input-domain classifier (d) better matches the true decision boundary.

influence of the reference domain increases and the decision boundary in the input domain is much improved compared to the true boundary.

In summary, Fig. 2 illustrates that unlike [12, 1, 4] our formulation remains applicable even when the decision boundary is non-linear. The effectiveness of our approach to cope with large inter-domain variations is demonstrated on the challenging mitochondria and ridge datasets, which we present next.

5.2 Transfer learning on Caltech 256

Following [2], we posed transfer-learning problems on the *edibles* and *vehicles* categories of the Caltech 256 dataset [17]. As was done in [2], for each pair of categories we performed two experiments by allowing each category to serve as either the reference or input domain; the results were then averaged across all pairs and ten random runs.

The experimental setup of [2] considers one to six positive samples in the input domain, making cross-validation for parameter selection (λ_2, C) uninformative, in which case our method underperforms [2]. However, if we choose the C parameter via cross-validation but then sweep through a valid range of λ_2 values, we obtain performance that is comparable or better than [2].

As the experiments in the following section demonstrate, [2] tends to underperform on the medical-imaging datasets. Such outcome seems to follow from the characteristics of the datasets, the experimental protocol and the algorithms being compared. First, the experimental protocol of [2] utilizes Caltech 256's *background* class which is not subject to the domain transformation. Although neither of the algorithms is specifically designed to exploit this property, the experimental setup may be a better match for the model-averaging approach of [2]. Second, by design, Caltech 256 is free from label noise. On the other hand, medical-imaging datasets typically contain label noise due to, e.g., ambiguity in the image evidence, disagreements between domain experts, or human error. The loss function employed in [2] tends to be more sensitive to this kind of noise. Since the LS SVM formulation of [2] has the property of selecting support

vectors both near and far from the decision boundary, the effect of label noise is detrimental to its performance. This can be observed in the medical imaging applications presented in the following sections.

5.3 Mitochondria Detection

Problem Definition. Mitochondria are membrane-enclosed organelles that play an important role in key cellular functions in addition to providing the cell with energy. Although mitochondria can be less than 10 μm, recent advances in Electron Microscopy (EM) have made it possible to reveal their shape and internal structure. It has also become clear that the amount of data contained in 3D EM stacks is overwhelming for processing by a human and may end up being woefully under-utilized. Being able to automatically segment mitochondria would allow neuro-scientists to gain new insights into degenerative disorders and perhaps develop life-saving cures.

State-of-the-art approaches to segmentation tend to require training with labeled data [18, 19]. Depending on the algorithm, class-label annotation may be required for each pixel, which means that acquiring such annotation in 3D becomes tremendously expensive, due both to the enormous size of the data and the fact that the operators have to be very skilled. Once a segmentation algorithm is trained, it is therefore essential to reduce the need for annotating subsequent 3D image stacks.

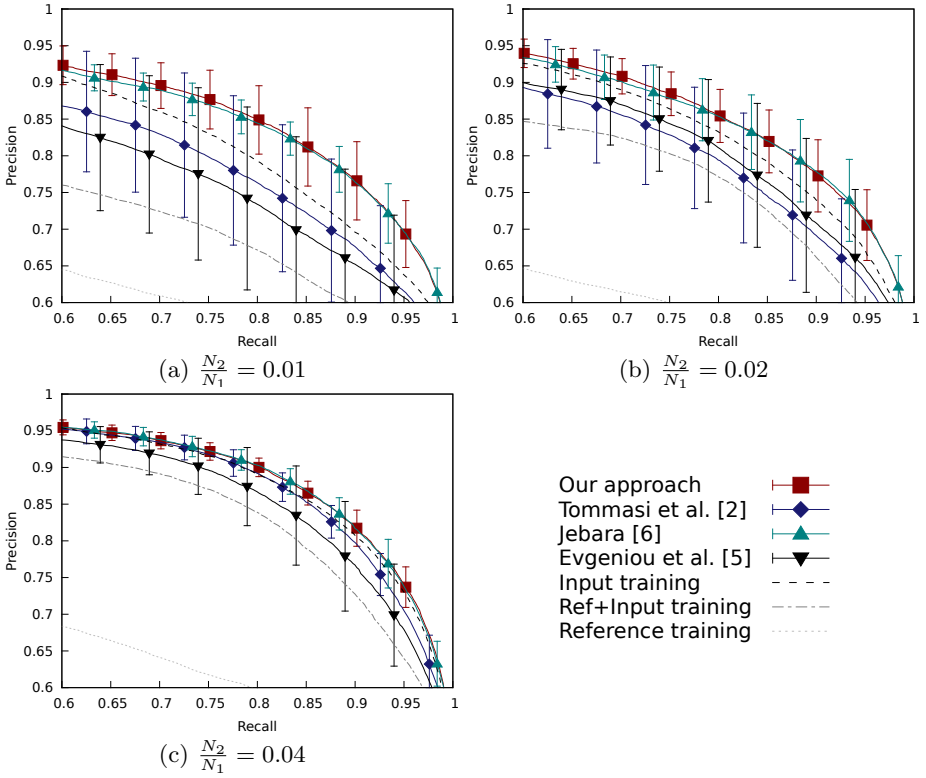
Unfortunately, the appearance of mitochondria varies considerably, as shown in Figs. 1(a,b). These variations may be due to where in the brain the samples are taken as well as from the acquisition settings. As a result, state-of-the-art segmentation algorithms trained on the reference domain may perform poorly on the input domain.

The transformation of features between the reference and the input domain cannot be easily normalized out, because, in most cases, the phenomenology that resulted in the domain change is not yet well understood. Without such knowledge, applying image processing operations, such as contrast adjustment, hurts the overall performance even further.

Experimental Setup. For our experiments we employ 3D image stacks from brain tissue acquired by milling the surface of the sample and imaging it with an electron microscope [20]. We treated the section from the *hippocampus* shown in Fig. 1(a) as the input domain and the section of the *striatum* shown in Fig. 1(b) as the reference domain.

We place ourselves in the context of the approach of [19], which works on supervoxels and classifies them as being *Inside a mitochondria* and *On the boundary of a mitochondria*. We therefore partition each 3D image stack into supervoxels using the publicly available code [2] described in [21] and compute histograms of grayscale frequencies for each supervoxel and its neighborhood, which yields a 20-dimensional feature vector per supervoxel.

[2] `http://ivrg.epfl.ch/supplementary_material/RK_SLICSuperpixels/index.html`



(a) $\frac{N_2}{N_1} = 0.01$ (b) $\frac{N_2}{N_1} = 0.02$

(c) $\frac{N_2}{N_1} = 0.04$

Fig. 3: [Best viewed in color] Precision-recall plot for the mitochondria transfer-learning experiment. Our transfer-learning approach performs comparably to [6] (but with a significantly-lower computational cost), and compares favorably to [2].

Results. For our transfer-learning experiment we consider *Boundary* and *Background* supervoxel labels. We work with 2,000 labelled examples in the reference domain and with between 1 and 4% of that amount in the input domain. Samples are randomly selected from both domains according to the specified quantities. The plots for the precision-recall curves for 10 random runs are shown in Fig. 3. Each curve represents the average, and the error bars denote one standard deviation. The plotted curves correspond to the performance of our proposed technique, the approaches of [2], [6], and [5]; also included are training purely either on the reference domain (*reference training*) or the input domain (*input training*) and a single SVM classifier that is fed the samples from both domains at once (*reference+input training*).

The plots confirm that the two domains differ significantly, as evidenced by the fact that the classifier trained on the reference domain (striatum) generalizes poorly on the input domain (CA1 hippocampus). In fact, its performance is so poor that its output is unlikely to be of any use in a segmentation algorithm.

We therefore turn our attention to comparison with the state-of-the-art transfer learning approaches.

Overall our approach achieves the best or comparable performance on these experiments. In particular, it attains comparable performance to [6], but at a fraction of the computational cost. When the amount of annotation is 1% and 2% compared to the reference domain, as in Fig. 3(a),(b), our precision-recall curve is slightly above [6] for some range of the recall values, and is overlapping [6] in Fig. 3(c). The remaining approaches tend not to fare well on this challenging dataset.

The approach of [2] under-performs when the amount of input-domain annotation is 1% and 2% compared to the reference domain. The performance improves as the amount of annotation increases to 4%, Fig. 3(c), but remains below ours for the range of the recall values that are likely to be useful in practice.

The approach of [5] does not fare well when the amount of input-domain annotation is 1%. Although it overtakes [2] when the fraction of the input-domain annotation is 2%, it lags behind all but two baselines when the amount of annotation is 4%.

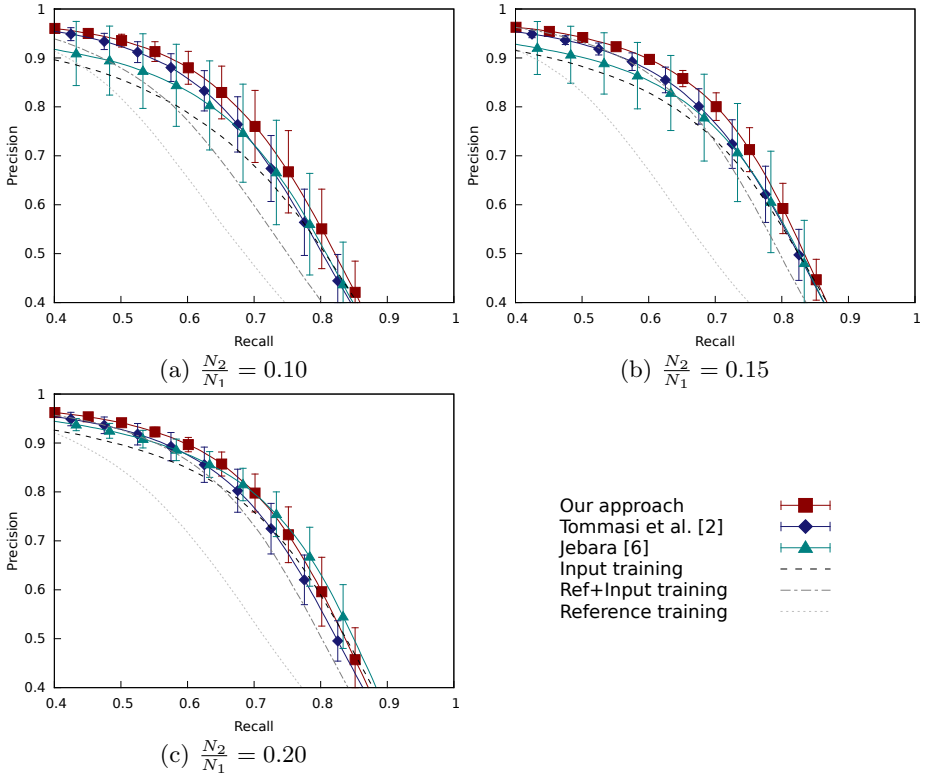
For the sake of completeness we summarize the comparison of our approach to the standard transfer-learning baselines: training only on the reference (or input) domains and combining the two domains.

When the amount of annotation is 1% compared to the reference domain, as in Fig. 3(a), the mixed classifier performs better than the one trained only on the reference domain. Interestingly, even with 1% of labeled samples, the classifier trained only on the input domain outperforms the mixed classifier. This could be explained by the drastic difference between the two domains: specialization to the reference domain hurts generalization on the input domain. By contrast our approach performs better than all the baselines and is good in absolute terms. The performance gain is particularly impressive when recall exceeds 0.5, which is the desired operating range for most applications.

When the amount of annotation is 2% compared to the reference domain, as seen in Fig. 3(b) the performance of the mixed classifier improves, but the negative influence of the reference domain remains pronounced. The performance of the classifier trained only on the input domain improves as well and remains superior to that of the mixed classifier. Nevertheless, our approach outperforms the other two in the useful operating range.

When the amount of annotation is 4% compared to the reference domain, the classifier trained only on the input domain performs well. The mixed classifier's performance also improves since there are now enough support vectors corresponding to the input domain. The benefit of transfer learning for this proportion of the input annotation becomes less pronounced, although the precision-recall curve for our algorithm is still somewhat above the others.

In summary, our approach achieves the best performance on the mitochondria datasets. It decisively outperforms all but one of the baselines in terms of accuracy, and it achieves comparable performance to [6]. Since [6] solves a SVM problem in an inner loop, its computational complexity can be high. In the mi-



(a) $\frac{N_2}{N_1} = 0.10$ (b) $\frac{N_2}{N_1} = 0.15$ (c) $\frac{N_2}{N_1} = 0.20$

Fig. 4: [Best viewed in color] Precision-recall plot for the ridge detection transfer-learning experiment. Our transfer-learning approach compares favorably to [2] in all three cases. When the fraction of the input-domain samples is low, i.e., (a),(b), our approach compares favorably to [6].

tochondria experiments the approach of [6] required as much as 300 seconds to train, while our approach required only 20 seconds. This speedup is essential for biomedical applications which are interactive or where the size of the reference dataset is large.

5.4 Ridge Detection

Problem Definition. To explore the behavior of our algorithm when confronted to an even more significant domain change, we use the data depicted by the bottom row of Fig. 1 in which the goal is to detect linear structures.

We use as our reference dataset a minimal intensity projection of Brightfield images, which were captured by an optical microscope from byocitin dyed brain tissue. The axons and dendrites appear as noisy black filaments to be delineated. We take our input dataset to be the publicly available DRIVE dataset [22] of 40 retinal scans such as those of Fig. 1. Here, the structures of interest are

the blood vessels which are different in appearance from the neurites of the Brightfield imagery. This is due, among other things, to the abrupt changes in contrast and background intensity.

Experimental Setup. It has been shown in [23] that feature vectors made of derivatives of order one to four computed using banks of steerable filters at different scales and orientations could be effectively used to classify individual pixels as being on the centerline of a tubular structure or not. We therefore compute such vectors in both domains and use them for our experiments.

The number of scales is set to three, following [23]. These scales are fixed for the reference and input datasets, in order to allow for a meaningful transfer learning task.

We work with 2,000 labelled samples in the reference domain, varying the amount of labelled examples used for training in the input domain.

Results. In Fig. 4 we present the results of our experiments on transfer-learning for ridge detection. Each precision-recall curve represents the average over ten random runs, and the error bars denote one standard deviation.

The plots in Fig. 4 confirm that the two domains differ significantly, as a reference-domain trained classifier performs poorly. The classifier trained on the mixture of the input and reference domains performs somewhat better, but is decisively outperformed by our transfer-learning approach. We now turn our attention to the comparison with the state-of-the art transfer learning baselines.

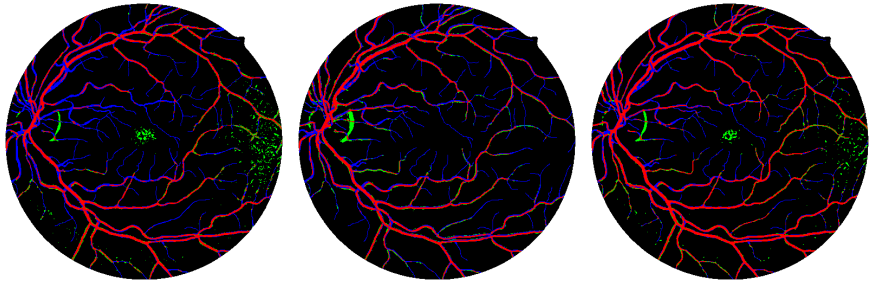
Overall, our approach fares well in all three cases. When the fraction of the input-domain examples is low compared to the reference domain Fig. 4(a,b), our approach clearly outperforms all competing baselines. When the fraction of input-domain examples increases to 20%, the performance of our method dominates all other baselines up to recall of 0.7, but then underperforms [6].

The approach of [2] under-performs on this dataset. When the fraction of input-domain examples is 10% and 15%, Fig. 4(a,b), its precision-recall scores tend to be comparable to [6], while remaining below those attained by our approach. When the fraction of input-domain examples is 20%, the technique of [2] is out-performed by both [6] and our approach.

To allow for a qualitative evaluation of the obtained performances, Fig. 5 presents the score images for three different classifiers for a given experimental run. Note that the reference domain provides useful classification information, as can be seen in Fig. 5a. Training the SVM on the input domain without enough data can lead to poor performance, as observed in Fig. 5b, depending on the complexity of the input domain and how well the available labelled data reflects this complexity. By contrast, our proposed approach overcomes these difficulties by fusing the data available from both domains, yielding visibly improved performance on the test data.

6 Conclusions

We proposed a transfer learning method based on sharing support vectors of the non-linear SVM classifiers. This new approach only requires a single additional



(a) Reference training (b) Input training (c) Our approach

Fig. 5: [Best viewed in color.] Ridge detection results. True positive (red), false positive (green) and false negative (blue) pixels obtained with the baseline approaches and with our method on a test image and $\frac{N_2}{N_1} = 0.10$. The threshold value was chosen at 0.02 false positive rate.

parameter compared to the standard SVM, which can be effectively optimized in a stratified fashion, as shown in our experiments.

We have demonstrated our approach on two challenging transfer-learning problems: classification of mitochondria supervoxels in FIB-EM 3D stacks from different parts of the brain and detection of tubular structures in brightfield projection images and the images of the retina. Although the degree of improvement over the baselines varies with the particular problem, our approach does better, or at worst the same as competing methods at a fraction of the computational cost.

7 Appendix

We now show that the $SVM_{S,T}$ of [8] is a special case of our proposed approach.

As shown in [5], their objective function for $\mathbf{w}_t = \mathbf{w}_0 + \mathbf{v}_t$

$$J(\mathbf{w}_0, \{\mathbf{v}_t\}) = \sum_{t=1}^{T} \sum_{i=1}^{m} \xi_{i,t} + \frac{\lambda_1}{T} \sum_{t=1}^{T} \|\mathbf{v}_t\|^2 + \lambda_2 \|\mathbf{w}_0\|^2, \tag{7}$$

can be re-written so that its regularization terms takes the form of

$$(\rho_1 + \rho_2) \sum_{t=1}^{T} ||\mathbf{w}_t||^2 - \rho_2 \frac{1}{T} \left\| \sum_{t=1}^{T} \mathbf{w}_t \right\|^2 \tag{8}$$

with

$$\rho_1 = \frac{1}{T} \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}, \quad \rho_2 = \frac{1}{T} \frac{\lambda_1^2}{\lambda_1 + \lambda_2}. \tag{9}$$

On the other hand, the objective in $SVM_{S,T}$ of [8] has the regularization term in the form of

$$||\mathbf{w}_S - \mathbf{w}_T||^2 = ||\mathbf{w}_S||^2 + ||\mathbf{w}_T||^2 - 2 < \mathbf{w}_S, \mathbf{w}_T > . \quad (10)$$

Setting $T = 2$ in Eq. 8 and comparing with Eq. 10 yields

$$\rho_2 = 2\ ,\ \rho_1 = 0 \quad \Rightarrow \quad \lambda_1 = 4,\ \lambda_2 = 0 \quad (11)$$

which is a special case of our formulation, since the penalty on the shared component $\mathbf{w}_0$ becomes zero.

References

1. Gopalan, R., Li, R., Chellappa, R.: Domain Adaptation for Object Recognition: An Unsupervised Approach. In: International Conference on Computer Vision. (2011)
2. Tommasi, T., Orabona, F., Caputo, B.: Safety in Numbers: Learning Categories from Few Examples with Multi Model Knowledge Transfer. In: Conference on Computer Vision and Pattern Recognition. (2010)
3. Jie, L., Tommasi, T., Caputo, B.: Multiclass Transfer Learning from Unconstrained Priors. In: International Conference on Computer Vision. (2011)
4. Aytar, Y., Zisserman, A.: Tabula Rasa: Model Transfer for Object Category Detection. In: International Conference on Computer Vision. (2011)
5. Evgeniou, T., Pontil, M.: Regularized Multi Task Learning. In: Conference on Knowledge Discovery and Data Mining. (2004)
6. Jebara, T.: Multitask Sparsity via Maximum Entropy Discrimination. Journal of Machine Learning Research **12** (2011)
7. Pan, S., Yang, Q.: A Survey on Transfer Learning. IEEE trans. on knowledge and data engineering **22** (2010)
8. Schweikert, G., Widmer, C., Scholkopf, B., Ratsch, G.: An Empirical Analysis of Domain Adaptation Algorithms for Genomic Sequence Analysis. In: Advances in Neural Information Processing Systems. (2008)
9. Kulis, B., Saenko, K., Darrell, T.: What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In: Conference on Computer Vision and Pattern Recognition. (2011)
10. Bruzzone, L., Marconcini, M.: Domain Adaptation Problems: a DASVM Classification Technique and a Circular Validation Strategy. IEEE Transactions on Pattern Analysis and Machine Intelligence **32** (2010)
11. Evgeniou, T., Micchelli, C., Pontil, M.: Learning Multiple Tasks with Kernel Methods. Journal of Machine Learning Research **6** (2005)
12. Obozinski, G., Taskar, B., Jordan, M.: Joint Covariate Selection and Joint Subspace Selection for Multiple Classification Problems. Statistical Computing **20** (2010)
13. Poggio, T., Mukherjee, S., Rifkin, R., Rakhlin, A., Verri, A.: Chapter 11. In: Uncertainty in Geometric Computations. Kluwer Academic Publishers (2001) 131–141
14. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
15. Gehler, P., Nowozin, S.: Let the Kernel Figure it Out: Principled Learning of Preprocessing for Kernel Classifiers. In: Conference on Computer Vision and Pattern Recognition. (2009)

16. Kovashka, A., Grauman, K.: Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: Conference on Computer Vision and Pattern Recognition. (2010)
17. Griffin, G., Holub, A., Perona, P.: Caltech-256 Object Category Dataset. Technical report, California Institute of Technology (2007)
18. Kaynig, V., Fuchs, T., Buhmann, J.: Neuron Geometry Extraction by Perceptual Grouping in ssTEM Images. In: Conference on Computer Vision and Pattern Recognition. (2010) 2902–09
19. Lucchi, A., Smith, K., Achanta, R., Knott, G., Fua, P.: Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks with Learned Shape Features. IEEE Transactions on Medical Imaging **31** (2011) 474–486
20. Knott, G., Marchman, H., Wall, D., Lich, B.: Serial Section Scanning Electron Microscopy of Adult Brain Tissue Using Focused Ion Beam Milling. Journal of Neuroscience **28** (2008) 2959–64
21. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Suesstrunk, S.: Slic Superpixels Compared to State-Of-The-Art Superpixel Methods. IEEE Transactions on Pattern Analysis and Machine Intelligence (2012)
22. Staal, J., Abramoff, M., Niemeijer, M., Viergever, M., van Ginneken, B.: Ridge Based Vessel Segmentation in Color Images of the Retina. IEEE Transactions on Medical Imaging **23** (2004) 501–509
23. Gonzalez, G., Fleuret, F., Fua, P.: Learning Rotational Features for Filament Detection. In: Conference on Computer Vision and Pattern Recognition. (2009) 1582–1589