Scalable Unsupervised Domain Adaptation for Electron Microscopy

Róger Bermúdez-Chacón[(✉)], Carlos Becker, Mathieu Salzmann,
and Pascal Fua

Computer Vision Lab, École Polytechnique Fédérale de Lausanne,
Lausanne, Switzerland
`roger.bermudez@epfl.ch`

Abstract. While Machine Learning algorithms are key to automating organelle segmentation in large EM stacks, they require annotated data, which is hard to come by in sufficient quantities. Furthermore, images acquired from one part of the brain are not always representative of another due to the variability in the acquisition and staining processes. Therefore, a classifier trained on the first may perform poorly on the second and additional annotations may be required. To remove this cumbersome requirement, we introduce an Unsupervised Domain Adaptation approach that can leverage annotated data from one brain area to train a classifier that applies to another for which no labeled data is available. To this end, we establish noisy visual correspondences between the two areas and develop a Multiple Instance Learning approach to exploiting them. We demonstrate the benefits of our approach over several baselines for the purpose of synapse and mitochondria segmentation in EM stacks of different parts of mouse brains.

Keywords: Domain Adaptation · Multiple instance learning · Electron microscopy · Synapse segmentation · Mitochondria segmentation

1 Introduction

Electron Microscopy (EM) can now deliver huge amounts of high-resolution data that can be used to model brain organelles such as mitochondria and synapses. Since doing this manually is immensely time-consuming, there has been increasing interest in automating the process. Many state-of-the-art algorithms [2,12,14] rely on Machine Learning to detect and segment organelles. They are effective but require annotated data to train them. Unfortunately, organelles look different in different parts of the brain as shown in Fig. 1. Also, since the EM data preparation processes are complicated and not easily repeatable, significant appearance variations can even occur when imaging the same areas.

In other words, the classifiers usually need to be retrained after each new image acquisition. This entails annotating sufficient amounts of new data, which is cumbersome. *Domain Adaptation* (DA) [11] is a well-established Machine Learning approach to mitigating this problem by leveraging information acquired
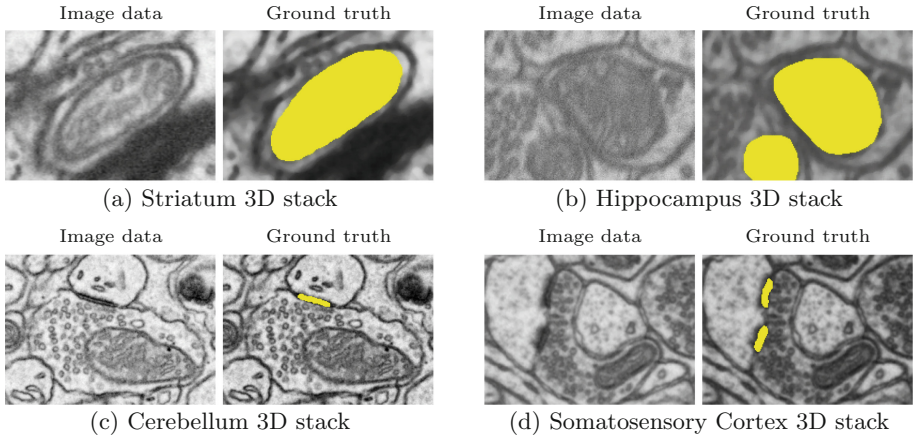
(a) Striatum 3D stack

(b) Hippocampus 3D stack

(c) Cerebellum 3D stack

(d) Somatosensory Cortex 3D stack

Fig. 1. Slices from four 3D Electron Microscopy volumes acquired from different parts of a mouse brain (annotated organelles overlaid in yellow). Note the large differences in appearance, despite using the same microscope in all cases.

when training earlier models to reduce the labeling requirements when handling new data. Previous DA methods for EM [3,17] have focused on the *Supervised DA* setting, which involves acquiring sufficient amounts of labeled training data from one specific image set, which we will refer to as the *source domain*, and then using it in conjunction with a small amount of additional labeled training data from any subsequent one, which we will refer to as the *target domain*, to retrain the target domain classifier.

In this paper, we go one step further and show that we can achieve *Unsupervised Domain Adaptation*, that is, Domain Adaptation without the need for *any* labeled data in the target domain. This has the potential to greatly speed up the process since the human expert will only have to annotate the source domain once after the first acquisition and then never again.

Our approach is predicated on a very simple observation. As shown in Fig. 2, even though the organelles in the source and target domain look different, it is still possible to establish *noisy* visual correspondences between them using a very simple metric, such as the Normalized Cross Correlation. By this, we mean that, for each labeled source domain sample, we can find a set of likely target domain locations of similar organelles. Not all these correspondences will be right, but some will. To handle this uncertainty, we introduce a Multiple Instance Learning approach to performing Domain Adaptation, which relies on boosted tree stumps similar to those of [3]. In essence, we use the correspondences to replace manual annotations and automatically handle the fact that some might be wrong.

In the remainder of this paper, we briefly review related methods in Sect. 2. We then present our approach in more detail in Sect. 3 and show in Sect. 4 that it outperforms other Unsupervised Domain Adaptation techniques.
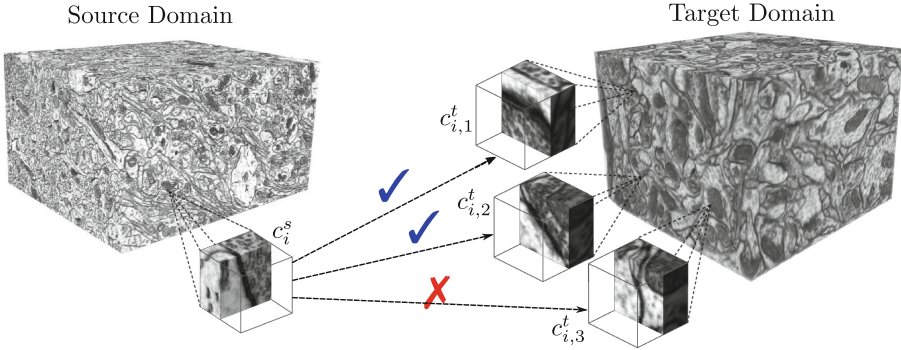
Fig. 2. Potential visual correspondences between an EM source stack (left) and a target stack (right) found with NCC. Our algorithm can handle noisy correspondences and discard incorrect matches.

2 Related Work

Domain Adaptation (DA) methods have proven valuable for many different purposes [11]. They can be roughly grouped in the two classes described below.

Supervised DA methods rely on the existence of partial annotations in the target domain. Such methods include adapting SVMs [5], projective alignment methods [4,20], and metric learning approaches [16]. Supervised DA has been applied to EM data to segment synapses and mitochondria [3], and to detect immunogold particles [17]. While effective, these methods still require manual user intervention and are therefore unsuitable for fully-automated processing.

Unsupervised DA methods, by contrast, do not require any target domain annotation and therefore overcome the need for additional human intervention beyond labeling the original source domain images. In this context, many approaches [1,10,15] attempt to transform the data so as to make the source and target distributions similar. Unfortunately, they either rely on very specific assumptions about the data, or their computational complexity becomes prohibitive for large datasets. By contrast, other methods rely on subspace-based representations [7,9], and are much less expensive. Unfortunately, as will be shown in the results section, the simple linear assumption on which they rely is too restrictive for the kinds of domain shift we encounter.

Recently, Deep Learning has been investigated for supervised and unsupervised DA [13,18]. These techniques have shown great potential for natural image classification, but are more effective on 2D patches than 3D volumes because of the immense amounts of memory required to run Convolutional Neural Nets on them. They are therefore not ideal to leverage the 3D information that has proven so crucial for effective segmentation [2]. By contrast, our approach operates directly in 3D, can leverage large amounts of data, and its computational complexity is linear in the number of samples.

3   Method

Our goal is to leverage annotated training samples from a *source domain*, in which they are plentiful, to train a voxel classifier to operate in a *target domain*, in which there are no labeled samples. Our approach is predicated on the fact that we can establish noisy visual correspondences from the source to the target domain, which we exploit to adapt a boosted decision stump classifier.

Formally, let f_{θ^s} be a boosted decision stump classifier with parameters θ^s trained on the source domain, where we have enough annotated data. In practice, we rely on gradient boosting optimization and use the spatially extended features of [2], which capture contextual information around voxels of interest. The score of such a classifier can be expressed as $f_{\theta^s}(\mathbf{x}^s) = \sum_{d=1}^{D} \alpha_d^s \cdot \operatorname{sign}(x_d^s - \tau_d^s)$, where $\boldsymbol{\alpha}^s = \{\alpha_1^s, \ldots, \alpha_D^s\}$ are the learned stump weights, $\Gamma^s = \{\tau_1^s, \ldots, \tau_D^s\}$ the learned thresholds, and $\mathbf{x}^s = \{x_1^s, \ldots, x_D^s\}$ the features selected during training. Given the corresponding features $\mathbf{x}^t$ extracted in the target domain, our challenge is to learn the new thresholds Γ^t for the target domain classifier $f_{\theta^t = \{\boldsymbol{\alpha}^s, \Gamma^t\}}$ without any additional annotations.

To this end, we select a number of positive and negative samples from the source training set $\mathcal{C}^s = \{c_1^s, \ldots, c_{N_c}^s\}$. For each one, we establish multiple correspondences by finding a set of k candidate locations in the target stack $\mathcal{C}_i^t = \{c_{i,1}^t, \ldots, c_{i,k}^t\}$ that visually resemble it, as depicted by Fig. 2.

In practice, correspondences tend to be unreliable, and we can never be sure that any $c_{i,j}^t$ is a true match for sample c_i^s. We therefore develop a Multiple Instance Learning formulation to overcome this uncertainty and learn a useful set of parameters Γ^t nevertheless.

3.1   Noisy Visual Correspondences

To establish correspondences between samples from both stacks, we rely on Normalized Cross Correlation (NCC). It assigns high scores to regions of the target domain with intensity values that locally correlate to a template 3D patch. We take these templates to be small cubic regions centered around each selected sample c_i^s in the source stack. Since the organelles can appear in any orientation, we precompute a set of 20 rotated versions of these patches. For each template, we compute the NCC at each target location for all 20 rotations and keep the highest one. This results in one score at every target location for each source template, which we reduce to the scores of the k locations with the highest NCC per source template via non-maximum suppression. Figure 3 shows some examples of the resulting noisy matches.

The intuition behind establishing correspondences is that, since we are looking for similar structures in both domains, they ought to have similar shapes even if the gray levels have been affected by the domain change. In practice, the behavior is the one depicted by Fig. 3. Among the candidates, we find some that do indeed correspond to similarly shaped mitochondria or synapses and some that are wrong. On average, however, there are more valid ones, which allows the robust approach to parameter estimation described below to succeed.
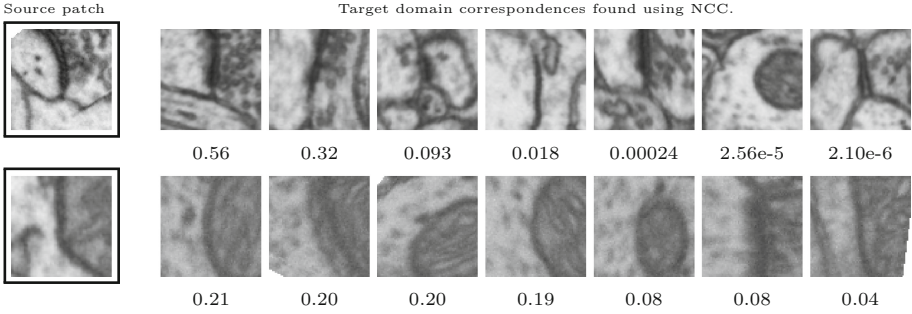
Source patch   Target domain correspondences found using NCC.

0.56   0.32   0.093   0.018   0.00024   2.56e-5   2.10e-6

0.21   0.20   0.20   0.19   0.08   0.08   0.04

Fig. 3. Examples of visual correspondences and their contributions to the gradient of the softmin function (Eq. 1) for synapses (top) and mitochondria (bottom).

3.2   Multiple Instance Learning

We aim to infer a target domain classifier given the source domain one and a few potential target matches for each source sample. To handle noisy many-to-one matches, we pose our problem as a Multiple Instance Learning (MIL) one.

Standard MIL techniques [19] group the training data into bags containing a number of samples. They then minimize a loss function that is a weighted sum of scores assigned to these bags. Here, the bags are the sets $\mathcal{C}_i^t$ of target samples assigned to each source sample c_i^s. We then express our loss function as

$$\hat{\Gamma}^t = \arg\min_{\Gamma^t} \frac{1}{|\mathcal{C}^s|} \sum_{c_i^s \in \mathcal{C}^s} \text{softmin}\left[\ell_{i1}, \ell_{i2}, \ldots, \ell_{ik}\right], \tag{1}$$

where $\ell_{ij} = L_\delta\left(f_{\theta^s}(c_i^s) - f_{\theta^t}(c_{i,j}^t)\right)$, L_δ is the Huber loss, and

$$\text{softmin}\left[\ell_1, \ldots, \ell_k\right] = -\frac{1}{r} \ln \frac{1}{k} \sum_{j=1}^{k} \exp(-r\ell_j) \tag{2}$$

is the log-sum-exponential, with $r = 100$ and $\delta = 0.1$ in our experiments. To find the parameters $\hat{\Gamma}^t$ that minimize the loss of Eq. 1, we rely on gradient boosting [8] and learn the thresholds one at a time as boosting progresses.

To avoid overfitting when correspondences do not provide enough discriminative information, we estimate probability distributions for the source and target thresholds τ_d^*. In particular, we assume that these thresholds follow a normal distribution $\tau_d^* \sim \mathcal{N}\left(\mu_{\tau_d}^*, (\sigma_{\tau_d}^*)^2\right)$, and estimate its parameters by bootstrap resampling [6]. For the source domain, we learn multiple values for each τ_d^s from random subsamples of the training data, and then take the mean and variance of these values. Similarly, for the target domain, we randomly sample subsets of the source-target matches, and minimize Eq. 1 for each subset. From these multiple estimates of τ_d^t we can compute the required means and variances.

Finally, we take $\hat{\tau}_d^t = \arg\max_\tau p(\tau_d^s = \tau)p(\tau_d^t = \tau)$, where $p(\tau_d^s)$ acts as a prior over the target domain thresholds: if the target domain correspondences produce high variance estimates, the distribution learned in the source domain acts as a regularizer.

4   Experimental Results

We test our DA method for mitochondria and synapse segmentation in FIBSEM stacks imaged from mouse brains, manually annotated (Fig. 1). We use source domain labels for training purposes and target domain labels for evaluation only.

For mitochondria segmentation, we use a $853 \times 506 \times 496$ stack from the mouse striatum as source domain and a $1024 \times 883 \times 165$ stack from the hippocampus as target domain, both imaged at an isotropic 5 nm resolution.

For synapse segmentation, we use a $750 \times 564 \times 750$ stack from the mouse cerebellum as source domain, and a $1445 \times 987 \times 147$ stack from the mouse somatosensory cortex as target domain, both at an isotropic 6.8 nm resolution.

4.1   Baselines

No adaptation. We use the model trained on the source domain directly for prediction on the target domain, to show the need for Domain Adaptation.

Histogram Matching. We change the gray levels in the target stack prior to feature extraction to match the distribution of intensity values in the source domain. We apply the classifier trained on the source domain on the modified target stack, to rule out that a simple transformation of the images would suffice.

TD Only. For each source example, we assume that the best match found by NCC is a true correspondence, which we annotate with the same label. A classifier is trained on these labeled target examples.

Subspace Alignment (SA). We test the method of [7]–one of the very few state-of-the-art DA approaches directly applicable to our problem, as discussed in Sect. 2. It first aligns the source and target PCA subspaces and then trains a linear SVM classifier. We also tested a variant that uses an AdaBoost classifier on the transformed source data to check if introducing non-linearity helps.

4.2   Results

For our quantitative evaluation, we report the Jaccard Index. Figure 4 shows that our method is robust to the choice of number of potential correspondences k; our approach yields good performance for k between 3 and 15. This confirms the importance of MIL over simply choosing the highest ranked correspondence. However, too large a k is detrimental, since the ratio of right to wrong candidates
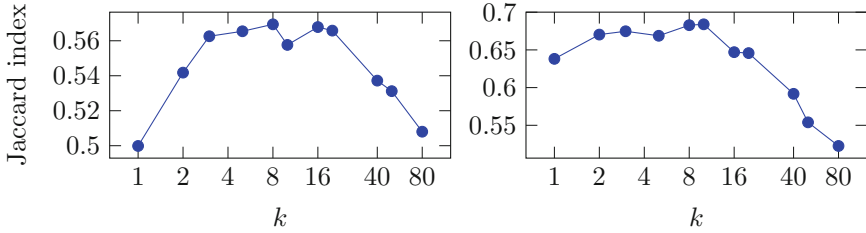
Fig. 4. Segmentation performance as a function of the number of candidate matches k used for Multiple Instance Learning, for the synapses (left) and mitochondria (right) datasets. Our approach is stable for a large range of values.

Table 1. Jaccard indices for our method and the baselines of Sect. 4.1.

	No adaptation	Histogram matching	TD only	SA [7] + Lin. SVM	SA [7] + AdaBoost	Ours ($k = 8$)
Synapse	0.22	0.32	0.39	0.13	0.39	**0.57**
Mitochondria	0.50	0.39	0.57	0.24	0.59	**0.62**

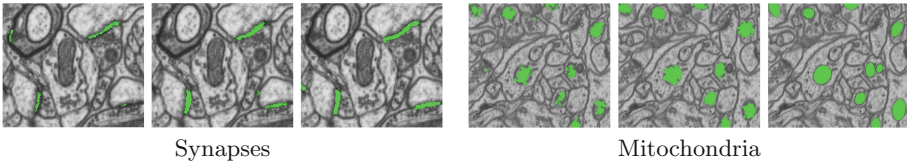Synapses   Mitochondria

Fig. 5. Detected synapses and mitochondria overlaid on one slice of the target domain stacks. In both cases, we display from left to right the results obtained without domain adaptation, with domain adaptation, and the ground truth.

then becomes lower. In practice, we used $k = 8$ for both datasets. Table 1 compares our approach to the above-mentioned baselines. Note that we significantly outperform them in both cases. We conjecture that the inferior performance of SA [7] is because our features are highly correlated, making PCA a suboptimal representation to align both domains.

The training time for the baselines was around 30 min each. Our method takes around 35 min for training. Finding correspondences for 10000 locations takes around 24 h when parallelized over 10 cores, which corresponds to around 81 s per source domain patch. While our approach takes longer overall, it yields significant performance improvement with no need for user supervision. All the experiments were carried out on a 20-core Intel Xeon 2.8 GHz.

In Fig. 5, we provide qualitative results by overlaying on a single target domain slice results with our domain adaptation and without. Note that our approach improves in terms of both false positives and false negatives.

5   Conclusion

We have introduced an Unsupervised Domain Adaptation method based on automated discovery of inter-domain visual correspondences and shown that its accuracy compares favorably to several baselines. Furthermore, its computational complexity is low, which makes it suitable for handling large data volumes. A limitation of our current approach is that it computes the visual correspondences individually, thus disregarding the inherent structure of the matching problem. Incorporating such structural information will be a topic for future research.

References

1. Baktashmotlagh, M., Harandi, M., Lovell, B., Salzmann, M.: Unsupervised domain adaptation by domain invariant projection. In: CVPR (2013)
2. Becker, C., Ali, K., Knott, G., Fua, P.: Learning Context Cues for Synapse Segmentation. TMI (2013)
3. Becker, C., Christoudias, M., Fua, P.: Domain adaptation for microscopy imaging. TMI **34**(5), 1125–1139 (2015)
4. Conjeti, S., Katouzian, A., Roy, A.G., Peter, L., Sheet, D., Carlier, S., Laine, A., Navab, N.: Supervised domain adaptation of decision forests: Transfer of models trained in vitro for in vivo intravascular ultrasound tissue characterization. Medical image analysis (2016)
5. Duan, L., Tsang, I., Xu, D.: Domain transfer multiple kernel learning. PAMI (2012)
6. Efron, B., Efron, B.: The jackknife, the bootstrap and other resampling plans, vol. 38. SIAM (1982)
7. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: ICCV (2013)
8. Friedman, J.: Stochastic Gradient Boosting. Computational Statistics & Data Analysis (2002)
9. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: CVPR (2012)
10. Heimann, T., Mountney, P., John, M., Ionasec, R.: Learning without labeling: domain adaptation for ultrasound transducer localization. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013. LNCS, vol. 8151, pp. 49–56. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40760-4_7
11. Jiang, J.: A Literature Survey on Domain Adaptation of Statistical Classifiers. Technical report, University of Illinois at Urbana-Champaign (2008)
12. Kreshuk, A., Koethe, U., Pax, E., Bock, D., Hamprecht, F.: Automated detection of synapses in serial section transmission electron microscopy image stacks. PloS one (2014)
13. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. In: ICML (2015)
14. Lucchi, A., Becker, C., Márquez Neila, P., Fua, P.: Exploiting enclosing membranes and contextual cues for mitochondria segmentation. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014. LNCS, vol. 8673, pp. 65–72. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10404-1_9
15. Pan, S., Tsang, I., Kwok, J., Yang, Q.: Domain adaptation via transfer component analysis. TNN **22**(2), 199–210 (2011)

16. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 213–226. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15561-1_16
17. Sousa, R.G., Esteves, T., Rocha, S., Figueiredo, F., Sá, J.M., Alexandre, L.A., Santos, J.M., Silva, L.M.: Transfer learning for the recognition of immunogold particles in TEM imaging. In: Rojas, I., Joya, G., Catala, A. (eds.) IWANN 2015. LNCS, vol. 9094, pp. 374–384. Springer, Heidelberg (2015). doi:10.1007/978-3-319-19258-1_32
18. Tzeng, E., Hoffman, J., Darrell, T., Saenko, K.: Simultaneous deep transfer across domains and tasks. In: ICCV (2015)
19. Viola, P., Platt, J., Zhang, C.: Multiple instance boosting for object detection. In: NIPS, pp. 1417–1424 (2005)
20. Wang, C., Mahadevan, S.: Heterogeneous domain adaptation using manifold alignment. In: IJCAI (2011)