Supervised domain adaptation of decision forests: Transfer of models trained *in vitro* for *in vivo* intravascular ultrasound tissue characterization[☆]

Sailesh Conjeti [a,1,*], Amin Katouzian [b,1,**], Abhijit Guha Roy [c], Loïc Peter [a], Debdoot Sheet [c], Stéphane Carlier [d], Andrew Laine [e], Nassir Navab [a,f]

[a] *Chair for Computer Aided Medical Procedures, Fakultät für Informatik, Technische Universität München, Germany*
[b] *IBM Almaden Research Center, San Jose, CA, USA*
[c] *Department of Electrical Engineering, Indian Institute of Technology Kharagpur, West Bengal, India*
[d] *Department of Cardiology, Hopital Ambroise Paré and Université de Mons, Mons, Belgium*
[e] *Department of Biomedical Engineering, Columbia University, New York, NY, USA*
[f] *Computer Aided Medical Procedures, Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA*

ARTICLE INFO

ABSTRACT

In this paper, we propose a supervised domain adaptation (DA) framework for adapting decision forests in the presence of distribution shift between training (source) and testing (target) domains, given few labeled examples. We introduce a novel method for DA through an error-correcting hierarchical transfer relaxation scheme with domain alignment, feature normalization, and leaf posterior reweighting to correct for the distribution shift between the domains. For the first time we apply DA to the challenging problem of extending *in vitro* trained forests (source domain) for *in vivo* applications (target domain). The proof-of-concept is provided for *in vivo* characterization of atherosclerotic tissues using intravascular ultrasound signals, where presence of flowing blood is a source of distribution shift between the two domains. This potentially leads to misclassification upon direct deployment of *in vitro* trained classifier, thus motivating the need for DA as obtaining reliable *in vivo* training labels is often challenging if not infeasible. Exhaustive validations and parameter sensitivity analysis substantiate the reliability of the proposed DA framework and demonstrates improved tissue characterization performance for scenarios where adaptation is conducted in presence of only a few examples. The proposed method can thus be leveraged to reduce annotation costs and improve computational efficiency over conventional retraining approaches.

1. Introduction

Atherosclerotic tissue characterization (TC) has been widely investigated by researchers over the past decade, particularly towards identification of vulnerable plaques. This is imperative during percutaneous coronary intervention (PCI) for selecting the most appropriate treatment strategy for stabilizing the plaque (Mintz, 2014). For this purpose, intravascular ultrasound (IVUS) has been recognized as the most prominent modality as it is easy to be deployed for intra-operative PCIs and more importantly provides cross-sectional images with adequate spatial resolution along with sufficient penetration. As a result, several TC algorithms were developed including texture based (Escalera et al., 2009; Taki et al., 2010; Katouzian et al., 2012a), radio-frequency based (Nair et al., 2002; Ohota et al., 2012; Sathyanarayana et al., 2009) and hybrid approaches (Seabra et al., 2011; Sheet et al., 2014). In this paper, the task of TC is to perform pixel-level probabilistic labeling of heterogeneous tissues *in vivo* for vascular plaques using IVUS, thus providing real-time tissue composition information to clinicians for better plaque assessment.

In general, the majority of the techniques were designed, developed, and validated upon *in vitro* collected data and directly deployed for *in vivo* application without taking into account the effect of dynamic factors like blood (Katouzian et al., 2008; Ciompi et al., 2010). However, the physics behind ultrasound image acquisition indicates that the signal transmission medium between the signal source and acoustic scatters significantly influences the
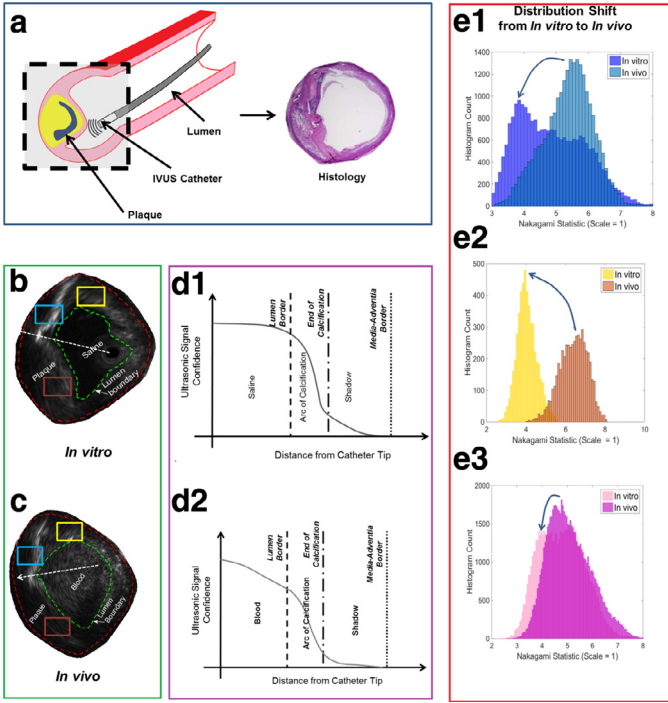
Fig. 1. Difference between *in vitro* and *in vivo* scenarios: IVUS provides a real-time cross-sectional image of the arterial wall and plaque micro-structure (schematic of acquisition in (a)). Due to availability of histological validation, TC algorithms are trained on *in vitro* labeled data (b – rectangular regions indicate calcified (blue), lipidic (yellow) and fibrotic (red)), which is collected from *post-mortem* hearts circulating saline in lumen. We use tissue-specific back-scattering signal statistics (e1 – calcified (violet (*in vitro*) and blue (*in vivo*)), e2 – lipidic (yellow (*in vitro*) and orange (*in vivo*))) and e3 – fibrotic (pink (*in vitro*) and magenta (*in vivo*))) and ultrasonic signal confidence (d1) for learning a TC model. Transitioning to *in vivo*, the presence of blood in the lumen (c – rectangular regions indicates calcified (blue), lipidic (yellow) and fibrotic (red)) and artery motion results in incoherently-attenuated IVUS signals, resulting in shift in distribution of signal backscattering statistics (e1–e3) and lowering of the ultrasonic signal confidence (d2). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

back-scattering signal characteristics (Culjat et al., 2010). This extends to the case of IVUS data acquisition, where the transmission media (blood for *in vivo* and saline for *in vitro*) induces changes in the tissue-specific back-scattering signal statistics and signal confidence values which are used to perform IVUS TC (illustrated in Fig. 1).

This is also supported by empirical observations based on ultrasound attenuation power law fit for whole blood components for frequency range of 0–70 MHz which has been reported as $0.0546f^{1.58}$ dB/cm for whole blood with speed of sound of 1590 (±2.8) m/s and that for deionized water (very similar to saline) as $0.00139f^2$ dB/cm with speed of sound of 1524 m/s , where f is frequency in MHz (Treeby et al., 2011). Due to these changes in physical properties between the two media, we observe variability among signal statistics along with incoherent signal attenuation, thus resulting in a decreased confidence in the information carried by the ultrasound signal (Karamalis et al., 2012). These factors induce changes in the joint distribution of tissue labels and features describing signal backscattering statistics and signal confidence which are employed for TC. Consequently, such changes create differences between the *in vitro* and *in vivo* domains which is known as *distribution shift* and is illustrated in Fig. 1.

The main bottleneck in developing a classifier for *in vivo* application is to construct a comprehensive and reliable *in vivo* dataset, which is often challenging if not infeasible due to: (1) lack of appropriate collectible samples, and (2) inconsistency amongst expert-defined labels (*due to non-availability of histological valida-*
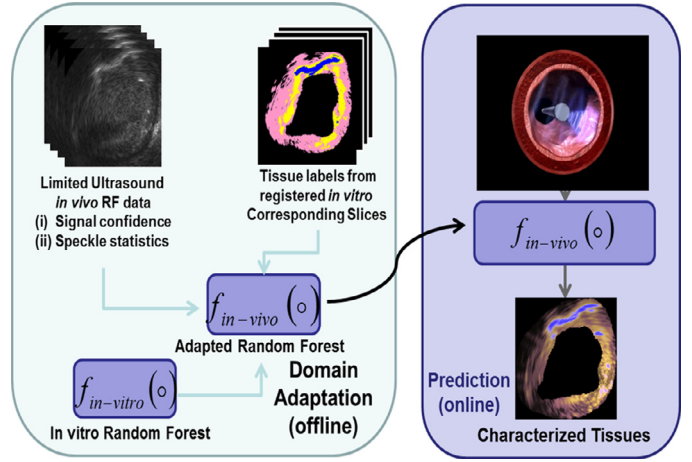
Fig. 2. Overview of the proposed method for domain adaptation of *in vitro* trained random forests for *in vivo* intravascular ultrasound TC.

tion). In such a scenario, where conventional histology assessment cannot be performed on living patients, it is effective to leverage the exhaustive *in vitro* dataset and suitably modify the models trained on it to provide a realistic solution for *in vivo* TC. We establish our framework on this premise and explore the challenges involved within *in vitro* to *in vivo* translation and propose to suitably modify the *in vitro* model for reliable *in vivo* TC despite limited availability of learning examples.

Supervised machine learning requires reliable labeled data for training and validating decision models. Often, it is assumed that the training and the testing data are from the same distribution (called a *domain*). In real world, this assumption often fails (*e.g.* due to dataset bias, domain difference, *etc.*; Torralba and Efros, 2011) and the learned models must adapt to the data in which they are deployed. Extending this argument to IVUS TC, models trained on *in vitro* data have to factor out the presence of uncontrolled features like blood and arterial motion, which induce changes in distribution of the IVUS backscattered signals.

In this paper, we present a novel technique for supervised domain adaptation of random forests. For the first time, we deploy it to adapt a reliably trained *in vitro* classifier (source) for *in vivo* (target) applications. Forests for characterizing under *in vitro* conditions are trained with ultrasound back-scattering statistical features from IVUS radio-frequency signals and signal confidence maps (as discussed in Sheet et al., 2014). Prior to presenting the detailed exposition of our work, we provide a quick insight into our domain adaptation technique to highlight the methodological contributions of this work (also illustrated in Fig. 2). We do this by answering three fundamental questions posed by Tommasi et al. (2014) that are necessary for defining any knowledge transfer method including domain adaptation, as follows:

1. *What to adapt?* In the proposed framework, we do model adaptation, where the source-trained decision forests are adapted to the target domain with a few labeled examples. The forest is adapted to be maximally discriminative on the target domain by adapting the split decision boundaries and leaf node posteriors from source instead of completely retraining them. The proposed method works for the most generic case of distribution shift between related domains. Here, we make no assumptions on the relationship of class-posterior probabilities or priors.

2. *How to adapt?* Adaptation of decision forests is performed treewise at the level of split and leaf nodes. To begin with, we cast the minimization of the global distribution shift between the domains as a domain alignment problem (Section 3.2). Following which, local domain differences are corrected in a top-down

fashion traversing through the tree structure. This is achieved through the proposed methods for sign-corrected local feature normalization and error-correcting hierarchical transfer relaxation at each split node (Section 3.5.1). At the leaf nodes, we propose adaptation *via* leaf posterior re-weighting and split to leaf node conversion to effectively handle the biases of source and target decision models (Section 3.5.2). Each of these methods is the major methodological contributions of the paper and is tailor-designed for adaptation of different components of a random forest, with few labeled target domain examples.

3. *When/How much to adapt?* In this paper, we perform model transfer based adaptation, wherein all split and leaf nodes need to be adapted to the target domain. However, with the error-correcting hierarchical transfer relaxation formulation, we regularize the amount of knowledge transfer between the source and target data. If the source data is highly sub-optimal to the target data (despite domain alignment and local normalization) due to high domain differences, the source boundary is increasingly relaxed and is corrected. Further, down the tree, at the levels of the leaf nodes, posterior predictive model re-weighting is locally adaptive and is suited to handle local domain differences like local co-variate shift in an effective fashion.

The rest of the paper is organized as follows. In Section 2, we discuss the state of the art in domain adaptation, transfer learning of decision trees as well as random forests, and present potential areas of applications in medical image computing. Section 3 discusses in detail the proposed mechanisms to train and adapt the forests. In Section 4, we extensively validate the proposed formulation for the target application of *in vivo* IVUS TC against the baselines of direct deployment of *in vitro* classifier, learning *in vivo* classifier with limited data, and joint domain-invariant learning. Finally, we conclude in Section 5, highlighting the major observations and conclusions and lay the ground for future investigations.

2. State of the art

Methods to mitigate the degradation of classifier performance due to distributional change between the training and the testing domains is an increasingly active area of research in the machine learning and computer vision communities (Shimodaira, 2000; Ben-David et al., 2010; Patel et al., 2015). In this context, popular approaches include transfer learning or multi task learning (Caruana, 1997), self-taught learning (Raina et al., 2007), semi-supervised learning (Chapelle et al., 2010), concept drift (Widmer and Kubat, 2011), multi view learning (Sun, 2013), and domain adaptation (Patel et al., 2015).

Intuitively, the aim of DA is to leverage knowledge (like decision models, datasets *etc.*) acquired from one or more *source* domains, and apply it to a closely related *target* domain, where there is no or few labeled data (Patel et al., 2015).

Depending on the availability of source and target domain data, DA is broadly categorized into four groups: (1) Supervised: large labeled source data and few labeled target data; (2) Unsupervised: large labeled source data and unlabeled target data; (3) Multi source: multiple source domains with both semi-supervised and unsupervised categories and (4) Heterogeneous: features between source and target domains are different (Patel et al., 2015). The current work falls under the category of supervised domain adaptation as we adapt forests for *in vivo* TC by leveraging *in vitro* data and models (source) along with few labeled *in vivo* examples (target). We also review the prior reported approaches towards transfer learning and domain adaptation of decision trees as well as forests and discuss their applications in medical image computing where these methods have increasing potential to be applied, yet less investigated.

2.1. Approaches for adaptation of decision trees and random forests

Semi-supervised learning on random-forests is closely related to DA as it leverages the available unlabeled data using a maximum margin approach which is optimized through deterministic annealing (Leistner et al., 2009). However, in this method, the labeled and unlabeled data are assumed to be from the same distribution. Further, won Lee and Giraud-Carrier (2007) proposed to perform transfer learning on decision trees in a semi-incremental fashion that uses a partially learned decision tree from a previous task instead of learning from scratch for a new task. Extending transfer learning of trees to forests, Goussies et al. (2014) proposed the mixed information gain criterion and label propagation to train trees using multi-task datasets in a joint fashion. In Goetz et al. (2012), forests were adapted to increase class-specificity through instance weighting and in Vezhnevets and Buhmann (2011), unsupervised DA was used for image categorization and semantic segmentation through joint-training of semantic texton forests. An alternative to adaptation or transfer learning of forests in presence of limited training data is to use decision jungles proposed by Shotton et al. (2013). It is a variant of conventional random forests that allows multiple paths between split and leaf nodes. This effectively reduces the number of nodes and thus avoids potential over-fitting due to limited training data. In contrast to the above methods for adaptation, we propose a generic DA method for random forests that effectively factors out domain differences like distribution shift without making any prior assumptions on domain separability.

2.2. Applications in medical image computing

DA and transfer learning are becoming increasingly popular in medical image computing applications. By learning with few or no labeled target examples, DA effectively mitigates the high labeling and data acquisition costs in medical data, which would have to be incurred if a large representative dataset were to be acquired for every domain evolution. Plausible scenarios where DA can be leveraged include upgrading of scanner platforms, in multi-center studies where image acquisition protocol may differ, new patient population, etc. Towards this end, prior applications and approaches reported in medical imaging literature include training from synthetically generated samples in ultrasound transducer localization (Heimann et al., 2015) and reducing annotation costs in microscopic images by leveraging labeled examples from closely related domains (Becker et al., 2015). Further, in Goetz et al. (2012), random forests were adapted to increase class-specificity through instance weighting under covariate shift assumption for brain tumor classification with limited labeled data and in Cheng et al. (2012) DA was employed for classifying Alzheimer's disease stages.

In van Opbroek et al. (2015), protocol-invariant models for supervised segmentation of brain MRI images were proposed for multi-center neuroimaging applications. In a multi-center transfer learning study undertaken by van Engelen et al. (2015) for carotid artery plaque component characterization, it was established that feature normalization and transfer learning can be effective for supervised learning algorithms to work across scanner platforms. These works demonstrate an increasing interest of the medical image computing community towards domain adaptation and transfer learning and this work is considered as an attempt in this general direction. In contrast to the above applications, for the first time, we adapt classifiers trained with *in vitro* conditions to be maximally discriminative during *in vivo* conditions like in translational use or clinical deployment.

Recently, leveraging deep convolutional neural networks for the task of pixel-level classification and segmentation are becoming increasingly popular and have also found applications in the

medical image computing domain (LeCun et al., 2015; Schmidhuber, 2015). In case of limited training examples, where training a deep network is difficult, one can leverage pre-trained deep networks trained for large-scale image classification tasks like imageNet (Krizhevsky et al., 2012) to extract features and employ them for the target task in a transfer learning framework. These features can also be used in conjunction with other classifiers like support vector machines, random forests for performing image segmentation. Such deep learning based approaches entail an advantage that the features are not hand-crafted but learnt from the data. Recently, supervised and unsupervised domain adaptation approaches have been extended to deep learning using domain interpolation and back propagation (Chopra et al., 2013; Ganin and Lempitsky, 2015). It must however be noted that ultrasound, unlike typical natural images, is stochastic in nature and is speckle based imaging modality. For successful extension to TC applications, deep learnt networks should be tailor-made to handle stochasticity and uncertainty observed in ultrasonic signals. However, as the current focus of this work is on adaptation of random forests, we do not delve further into deep learning but identify that deep learning paradigms hold a promising future.

3. Methodology

3.1. Domain adaptation for in vivo TC

For the proposed application, we employ a probabilistic TC framework developed for characterizing co-located heterogeneous tissues in IVUS through joint learning of tissue-specific statistical physics attributes and signal confidence (ultrasonic attenuation). This is adapted from the Stochastic Driven Histology framework developed by Sheet et al. (2014). The factors influencing the ultrasonic back-scattering such as the multi-scale statistical distribution estimators and signal attenuation metrics are incorporated into the TC model as predictors. Due to the purely stochastic nature of ultrasonic signals (especially radio frequency data) and the co-located heterogeneity of constituent tissues of atherosclerotic plaques, we use an analytic model of back-scattering (multi-scale Nakagami distribution) as proposed in the seminal work by Shankar (2001). This model effectively encompasses varied conditional distribution models (Rayleigh, k-distributed and Rician models for ultrasound backscattering) into a single statistical distribution model. Using features derived from this model, we obtain signatures that are highly distinctive for varying acoustic scatterer sizes, density and distribution. This is particularly suitable for TC as different tissues constituting the plaque have different scatterer properties and composition. In addition to considering ultrasonic backscattering derived features, we also incorporate ultrasonic signal confidence derived using random walks based formulation proposed by Karamalis et al. (2012). The confidence of the received ultrasonic signal plays a crucial role in incorporating signal uncertainty into the TC framework. This is inspired by the observation that the fidelity of ultrasound signal decreases as it penetrates further into the tissues and the information carried by peripheral tissues is less reliable for TC. The types of tissues considered include fibrotic, lipidic, and calcified tissues. We manually demarcate the lumen and media-adventitia borders to delineate the plaque region. Regions beyond the media-adventitia borders and within the lumen are deemed to belong to non-tissue classes. The evaluation for TC is however performed only on plaque region as ultrasonic signal beyond the media adventitia border has very low confidence due to high back-scattering by the plaque tissues and TC results outside the arterial wall is obsolete.

From the perspective of DA, we define the in vitro acquired data as the source domain and the limited in vivo dataset as the target domain. Extracting features through multi-scale prediction of

Table 1
List of important notations.

Source (s)	Target (t)	Description
X_{src}, Y_{src}	X_{tar}, Y_{tar}	Data, tissue labels
$\mathcal{P}_{src}$	$\mathcal{P}_{tar}$	Projection matrices into principal component space
	$\mathcal{M}_a$	Target to source alignment matrix
	ΔD	Measure of domain difference
h_{src}	h_{tar}	Hypothesis (tree)
S_{src}	S_{tar}	Split node
$\mathcal{D}^S_{src}$	$\mathcal{D}^S_{tar}$	Data reaching split node
$\mu^S_{src}, \sigma^S_{src}$	$\mu^S_{tar}, \sigma^S_{tar}$	Feature-wise mean and standard deviation of $\mathcal{D}^S_{src}$ and $\mathcal{D}^S_{tar}$
	ϕ	Oblique split function
θ^S_{src}	θ^S_{tar}	Parameters of split function
$\alpha^S_{src}, \alpha^{S0}_{src}$	$\alpha^S_{tar}, \alpha^{S0}_{tar}$	Oblique split coefficients and intercept
	$\mathcal{D}^S_{CR}, \mathcal{D}^S_{CL}$	Data split to left and right child nodes (CL, CR)
	$\mathcal{I}$	Information gain
L_{src}	L_{tar}	Leaf node
$\mathcal{D}^L_{src}$	$\mathcal{D}^L_{tar}$	Data reaching leaf node
$\mathcal{R}^L_{src}$	$\mathcal{R}^L_{tar}$	Subspaces encoded by leaf node
p^L_{src}	p^L_{tar}	Leaf posterior model
	Γ	Transfer relaxation parameter
	$\mathcal{S}$	Source-regularized hypothesis space
	$\Gamma_c, \Gamma_m, \Gamma_f$	Transfer relaxation (coarse, medium and fine)
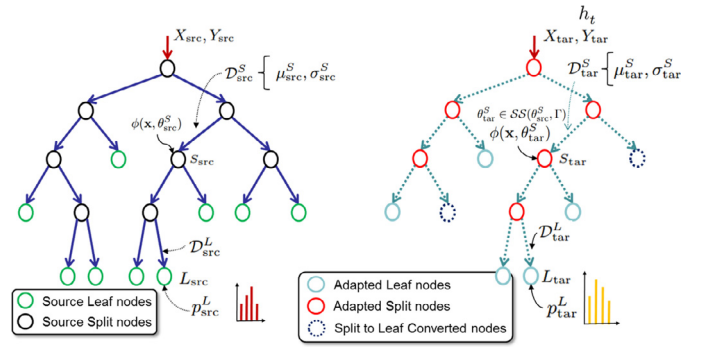
Fig. 3. Schematic illustrating major notations used in the paper. The source tree hypothesis h_{src} is trained with dataset X_{src}, Y_{src}. The tree comprises of set of split nodes (S_{src}), with an oblique split function (denoted as $\phi(\mathbf{x}, \theta^S_{src})$). The splitting terminates in leaf nodes L_{src}, which have a posterior predictive model p^L_{src} associated with them. Domain adaption involves generating the adapted tree hypothesis h_{tar} using h_{src} and few labeled target examples X_{tar}, Y_{tar}. The split and leaf nodes are modified to generate adapted nodes S_{tar} and L_{tar}, with modified split function $\phi(\mathbf{x}, \theta^S_{tar})$ and predictive models p^L_{tar}, respectively. During adaptation, split nodes of the source tree that are not reached or do not possess sufficient target samples for reliable adaptation are pruned and converted to leaf nodes.

Nakagami parameters and signal confidence (as discussed in Sheet et al., 2014), we constitute a 57 dimensional feature vector for each ultrasound resolution cell. Further, we align the feature spaces of in vivo dataset to the in vitro datasets using domain alignment method to make them comparable using the method discussed in Section 3.2. Following which, the TC random forest classifier trained on in vitro data is adapted to be maximally discriminative on the target domain by correcting for local domain shifts and errors in the decision boundaries using the adaptation mechanisms presented in Section 3.5. This adapted classifier is the final outcome of the proposed DA method. In the subsequent sections, we discuss in detail the individual steps proposed for the DA method.

Prior to presenting the mathematical formulation, we list down the notations used in this section in Table 1 and support it with Fig. 3, which schematically illustrates them with respect to source and target (or adapted) tree.
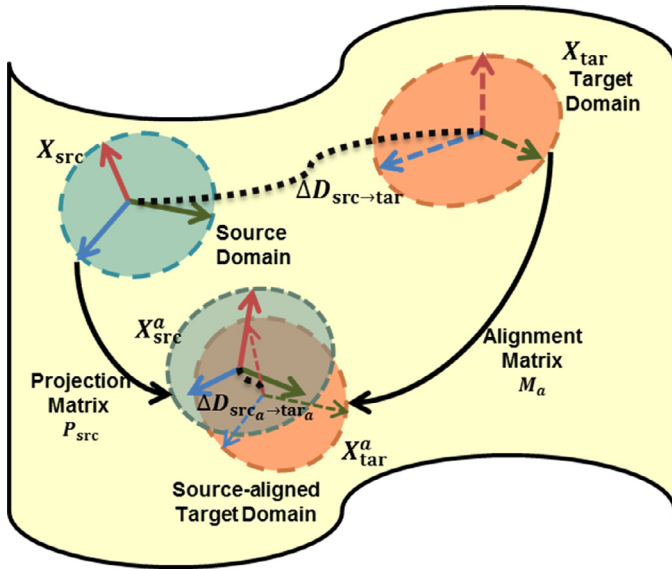
Fig. 4. Schematic illustrating minimization of domain difference through domain alignment as a preprocessing step.

3.2. Preprocessing: global domain alignment through robust Principal Component Analysis

The performance of the DA algorithm is optimal if the joint distributions of the target and source domains are *closely related*. In the context of the present work, the *in vivo* and *in vitro* domains are closely related as the tissue composition is invariant to the media in the lumen. As suggested by Ben-David et al. (2007), a reduction in domain difference between two domains is required for successful adaptation as it improves comparability between the domains. Towards this end, subspace-based domain adaptation techniques have shown promising results to tackle global domain shifts between the source and target domains (Pan et al., 2008; Fernando et al., 2013). We utilize the method proposed by Fernando et al. (2013) to perform PCA-based domain alignment, by transforming the domains to a common latent space where the global domain differences are minimal. The technique used in this paper differs minorly from Fernando et al. (2013) in the following aspects: (1) we use robust PCA computed with Grassmanian averages, which is robust to outliers (Hauberg et al., 2014), and (2) we align the target space to the source subspace to facilitate DA without the need for retraining the source classifier (explained in Section 3.4). Fig. 4 illustrates the scheme of minimizing the domain differences ($\Delta D_{\text{src} \to \text{tar}}$ to $\Delta D_{\text{src}_a \to \text{tar}_a}$ post alignment) through the proposed alignment scheme (domain difference measure ΔD is discussed in Section 4.5.1).

Exploiting the global covariance statistical structure of the two domains, the source domain data $X_{\text{src}} \in \mathbb{R}^N$ and target domain data $X_{\text{tar}} \in \mathbb{R}^N$ are projected into their respective eigenvector subspaces through linear projection matrices $\mathcal{P}_{\text{src}} = \text{rPCA}(X_{\text{src}}, d)$ and $\mathcal{P}_{\text{tar}} = \text{rPCA}(X_{\text{tar}}, d)$ estimated using robust Principal Components analysis (rPCA) for the subspace dimension d (Hauberg et al., 2014). Following this, the target subspace is aligned to the source subspace ($X_{\text{src}}^a (\in \mathbb{R}^d) = X_{\text{src}} \mathcal{P}_{\text{src}}$) to generate the source-aligned target space $X_{\text{tar}}^a (\in \mathbb{R}^d)$. As shown in Fernando et al. (2013), the linear projection matrix that aligns the basis vector of the two domains, which in turn minimizes the Bregman matrix divergence between the two domains is defined by $\mathcal{M}_a = \mathcal{P}_{\text{tar}} \mathcal{P}_{\text{tar}}^T \mathcal{P}_{\text{src}}$ and $X_{\text{tar}}^a = X_{\text{tar}} \mathcal{M}_a$.

Domain alignment *via* rPCA is intrinsically regularized and it must be noted that if a target basis vector is orthogonal to all the source basis vectors, it is effectively ignored and vectors that are well aligned with the source basis vectors are given a higher weight. The theoretical upper-bound of the subspace dimensionality for achieving consistent alignment between the source and the target domains is discussed in Fernando et al. (2013). It must be noted that domain alignment works as a preprocessing step that aligns the two domains in a global way. At a local level, the sub-domains are typically comparable but not perfectly aligned (hence the need for adjusting decision boundaries as proposed later in Section 3.5). Domain alignment also helps handle correlated features by decorrelating them in the latent subspace, which might otherwise be detrimental towards learning. Henceforth, for notational simplicity, we use X_{src} and X_{tar} instead X_{src}^a and X_{tar}^a to refer to the aligned source and target subspaces respectively, unless mentioned otherwise.

3.3. Domain adaptation of decision forests

Decision forests are inherently multi-class classifiers, which effectively avoids training multiple binary *one vs. rest* classifiers for multi class problems like IVUS TC. Further, they are fast to train, test, adapt, and are effectively non-parametric. Decision forests tend to reduce the variance of the classifier through ensemble averaging of the results of independently learnt decision trees, thus, leading to better generalizability. In the particular cases of active and incremental learning with few examples, Salperwyck and Lemaire (2011) demonstrated through exhaustive validations on public benchmarked datasets that random forests with sufficient number of base learners in the decision ensemble is favorable in contrast to classifiers like support vector machines (with linear and radial basis function kernels), naïve Bayes (supervised and unsupervised), logistic regression, and tree-based classifiers (like C4.5, J48 and ADTree) to name a few. This premise is further extendable to domain adaptation with a few examples due to its similarity to incremental learning and thus supports the choice of decision forests for this application. In this work, we propose to use oblique random forests, which use oblique decision splits instead of traditional axis-aligned decision boundaries at split nodes. In addition to the above advantages that decision forests entail, oblique forests are superior in the following aspects: (a) ability to separate distributions that lie between the coordinate axes with a single multivariate split, which might have required deep nested axis-aligned splits otherwise, and (b) less bias of the learnt decision trees to the geometrical constraints imposed by the base learner (Menze et al., 2011). Meanwhile, in case of domain adaptation under distribution shift with the proposed method, oblique forests offer increasing degrees of freedom for adaptation, which could be very useful if the distribution shift happens between the coordinate axes.

Motivated by the above advantages, we present a framework for performing supervised DA of decision forests with few examples from a distribution-shifted target domain. The final goal of the proposed framework is to estimate the class posterior probability $p_{\text{tar}}(\mathbf{y}|\mathbf{x})$ for a previously unseen sample $\mathbf{x}$ conditioned on a set of few target training (or *adapting*) exemplars X_{tar} with annotated labels Y_{tar}. Towards this end, we leverage the large labeled source dataset X_{src} with labels Y_{src} along with models trained on it. Under distribution shift, the joint distribution on the source and target domains are different *i.e.* $p_{\text{src}}(\mathbf{x}, \mathbf{y}|X_{\text{src}}, Y_{\text{src}}) \neq p_{\text{tar}}(\mathbf{x}, \mathbf{y}|X_{\text{tar}}, Y_{\text{tar}})$. However, the feature and label space remain the same between the two domains, while the decision functions differ *i.e.* a particular instance may be labeled differently depending on the domain. The key idea is to infer a good approximation to $p_{\text{tar}}(\mathbf{x}, \mathbf{y})$ by leveraging the knowledge of the source decision functions from $p_{\text{src}}(\mathbf{x}, \mathbf{y})$ and labeled target examples.

3.4. Primer on random forest – training in source domain

Starting with the source domain, let $\mathcal{H}$ denote the hypothesis space. As a classification model, we use an ensemble of hypotheses conditioned on the source training dataset $\mathcal{D}_{src} = (X_{src}, Y_{src})$ defined on $\mathbb{R}^d \times \mathcal{Y}$, where $\mathcal{Y}$ is the label space. For a previously unseen source sample $\mathbf{x}$, the posterior probability $p(\mathbf{y}|\mathbf{x})$ is defined in the following sense:

$$p_{src}(\mathbf{y}|\mathbf{x}, X_{src}, Y_{src}) = \int_{h_{src} \in \mathcal{H}} p(\mathbf{y}|\mathbf{x}, h_{src}) p(h_{src}|X_{src}, Y_{src}) dh_{src} \quad (1)$$

where h_{src} is a hypothesis conditioned on (X_{src}, Y_{src}) generated from $\mathcal{H}$. Introducing the concept of bootstrapping - bagging and random subspace selection (Rodner and Denzler, 2009; Criminisi et al., 2012), we use Bayesian model averaging, which is approximated through a simple Monte Carlo estimation using N independently generated hypotheses. This is expressed as follows:

$$p_{src}(\mathbf{y}|\mathbf{x}, X_{src}, Y_{src}) = \frac{1}{N} \sum_{k=1}^{N} p(\mathbf{y}|\mathbf{x}, h_{src}^k(\widetilde{X_{src}^k}, \widetilde{Y_{src}^k})) \quad (2)$$

where h_{src}^k minimizes the source domain prediction error over selected subset $(\widetilde{X_{src}^k}, \widetilde{Y_{src}^k})$

$$h_{src}^k = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \, \mathbb{E}_{(\mathbf{x},\mathbf{y}) \subset (\widetilde{X_{src}^k}, \widetilde{Y_{src}^k})} \mathcal{P}(h(\mathbf{x}) \neq \mathbf{y}) + \mathcal{C}(h) \quad (3)$$

where $\mathcal{P}$ is the misclassification penalty function and $\mathcal{C}$ is the model complexity parameter (to prevent over-fitting like extremely deep trees). As we use randomized node optimization to find optimal hypothesis, $\mathcal{C}$ is controlled by configuration parameters (Θ_{src}) like minimum number of samples required to define a split node (minSplit). If minSplit were to be set to 1 sample, the resultant tree would grow extremely deep, thus becoming highly biased towards the training data. This can result in potential over-fitting and therefore reducing its generalizability. Optimally choosing minSplit would thus help prevent extremely deep trees and control model complexity $\mathcal{C}$.

The goal is to find the source decision forest, which is an ensemble of numTrees($= N$) independent hypotheses say $\mathcal{H}_{src} = \{h_{src}^1, \ldots, h_{src}^k, \ldots, h_{src}^{numTrees}\}$. Each of these decision tree hypothesis, say h_{src}^k is described by set of split nodes (with each node denoted as S_{src}) and a set of leaf nodes (each denoted as L_{src}). In case of oblique forests, these are independently trained multivariate decision trees, with oblique decision boundaries are ceach split node. For notational simplicity, we subsequently ignore the tree index k and wide tilde in h_{src}^k, ($\widetilde{X_{src}^k}$ and $\widetilde{Y_{src}^k}$) and refer to the source domain bagged subset as (X_{src}, Y_{src}) and hypothesis trained on it as h_{src}, unless mentioned otherwise.

Each tree is trained independently, introducing a certain level of randomness through bagging and random subspace selection as shown in Eq. (3). This is done so that the resultant hypotheses are decorrelated, and thus improve the generalization of the forests.

3.4.1. Split node

Let the training data that reaches split node S_{src} be denoted as $\mathcal{D}_{src}^S$ Starting with the root node, each feature of the data reaching a particular node is locally normalized with feature-wise mean μ_{src}^S and positive standard deviation σ_{src}^S, that are estimated locally from $\mathcal{D}_{src}^S$. This is done to handle variables of different dynamic ranges effectively. Thus, the oblique multivariate split $\phi(\mathbf{x}, \theta_{src}^S)$ is parameterized by θ_{src}^S which is a set of parameters describing the oblique split. Here, $\theta_{src}^S = \{\mu_{src}^S, \sigma_{src}^S, \alpha_{src}^S, \alpha_{src}^{S0}\}$, where α_{src}^S are the coefficients of individual features constituting the oblique split and α_{src}^{S0} is the intercept. Using this parameter set, the oblique split $\phi(\mathbf{x}, \theta_{src}^S)$ is formulated as below:

$$\phi(\mathbf{x}, \theta_{src}^S) = \left(\frac{\mathbf{x} - \mu_{src}^S}{\sigma_{src}^S} \right). \alpha_{src}^S - \alpha_{src}^{S0} \quad (4)$$

In this work, we use randomized node optimization (Criminisi et al., 2012) generating a family of candidate splits ($\mathcal{F}_{src}^S$), where each split (say $\theta_c \in \mathcal{F}_{src}^S$) is bivariate and assigned randomly generated coefficient values (say α_c computed from a parameter hypersphere of radius 1 centered at the origin (i.e. $\sqrt{\sum |\alpha_c|.^2} = 1$). The intercept α_c^0 is generated as a random value between the minimum and maximum of $\left(\frac{\mathbf{x} - \mu_{src}^S}{\sigma_{src}^S} \right)$. α_c. The coefficients are standardized by normalizing them to make their l_2 norm $= 1$ (i.e. $\sqrt{|\alpha_c|.^2 + |\alpha_c^0|^2} = 1$). The candidate split that maximizes the information gain (given by Eq. (6)) is assigned to the split node in which:

$$\theta_{src}^S = \operatorname{argmax}_{\theta_c \in \mathcal{F}_{src}^S} \mathcal{I}(\mathcal{D}_{src}^S, \theta_c) \quad (5)$$

The information gain over the dataset $\mathcal{D}_{src}^S$ for a candidate parameter set θ_c is computed as:

$$\mathcal{I}(\theta_c | \mathcal{D}_{src}^S) = \mathcal{H}(\mathcal{D}_{src}^S) - \sum_{i \in \{CL, CR\}} \frac{|\mathcal{D}_{n_{src}^i}|}{|\mathcal{D}_{src}^S|} \frac{\mathcal{H}(\mathcal{D}_{n_{src}^i})}{\mathcal{H}(\mathcal{D}_{src}^S)} \quad (6)$$

where dataset $\mathcal{D}_{src}^S$ is split into left and right subsets $\mathcal{D}_{CL}^S$ and $\mathcal{D}_{CR}^S$ by the split function as follows:

$$\mathcal{D}_{CL}^S = \{(\mathbf{x}, \mathbf{y}) | (\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{src}^S \wedge \phi(\mathbf{x}, \theta_c) \leq 0\} \quad (7)$$

$$\mathcal{D}_{CR}^S = \{(\mathbf{x}, \mathbf{y}) | (\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{src}^S \wedge \phi(\mathbf{x}, \theta_c) > 0\} \quad (8)$$

and $\mathcal{H}(S)$ is the entropy of set S, which is calculated as: $\mathcal{H}(S) = -\sum_{\forall y \in \mathcal{Y}} p_y \log(p_y)$ where p_y is the probability of class label y in any training set S. The pseudo-code for generating oblique splits (GenSplit) is discussed in Algorithm 1 in Appendix A.

Such a randomized node optimization along with bagging and randomized subspace selection ensures that the individual trees are de-correlated, which aids in improved generalization. This recursive splitting and training continues till the constraints imposed by the source domain configuration parameters Θ_{src} (like the depth of the tree treeDepth, minimum number of samples for splitting minLeaf) are satisfied or if all the samples reaching a node belong to the same class.

3.4.2. Leaf node

Let a terminated split node in tree k leads to a leaf node L_{src}, with a predictive model associated.

During training in the source domain, let $\mathcal{D}_{src}^L$ represent the set of samples reaching the leaf node. The subset of $\mathcal{D}_{src}^L$ with class $\mathbf{y} = y$ is given by $\mathcal{D}_{src}^L(\mathbf{y} = y)$. The predictive model $p_{src}^L(y|\mathbf{x})$ at node L_{src} acts as a posterior estimate of class probability of label $y \in \mathcal{Y}$ given a previously unseen test samples $\mathbf{x}$ reaching that leaf node. It is the normalized histogram of labels in $\mathcal{D}_{src}^L$ and is estimated as follows:

$$p_{src}^L(y|\mathbf{x}) = \frac{|\mathcal{D}_{src}^L(\mathbf{y} = y)|}{|\mathcal{D}_{src}^L|} \quad (9)$$

3.5. Domain adaptation to target domain

The task of supervised domain adaptation is to adapt each reliably trained source hypothesis h_{src} to be maximally discriminative on a closely-related target domain. We propose to domain adapt each hypothesis independently to generate the corresponding adapted hypothesis h_{tar}. In the context of decision forests, this implies adapting the split nodes and the leaf nodes of each decision tree. It is important to preserve the de-correlated nature of the trees even after adaptation. For this, as an input for adaptation, we perform bagging on the target domain data $\mathcal{D}_{tar} = (X_{tar}, Y_{tar})$ to obtain $(\widetilde{X_{tar}^k}, \widetilde{Y_{tar}^k})$. Domain adapting h_{src} is posed as minimizing the
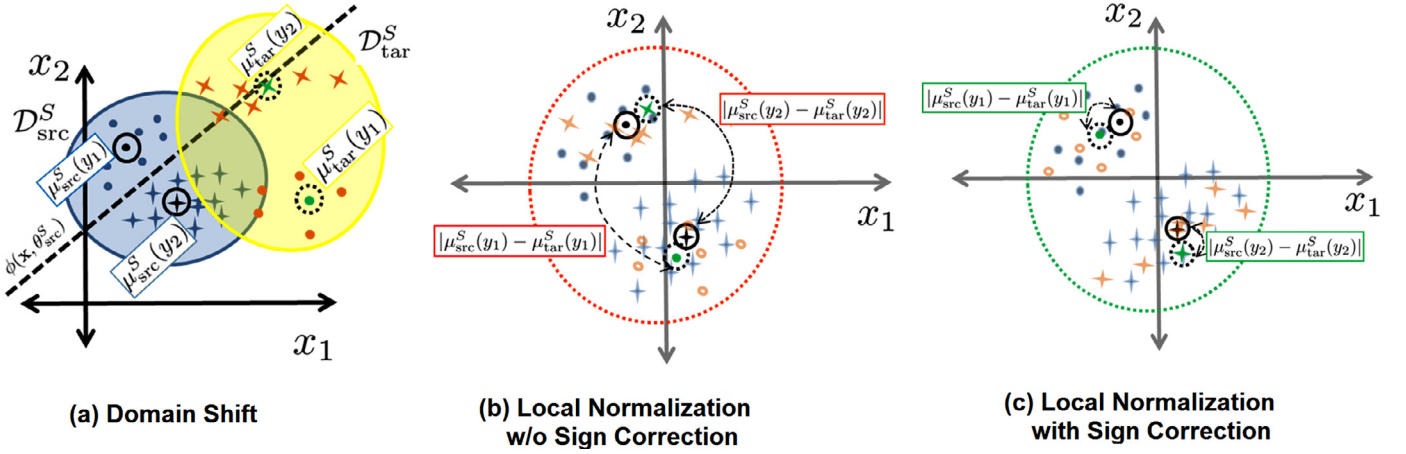
Fig. 5. Schematic illustrating the effect of sign correction during local node normalization. (a) At the level of each split node, the source and target domains may exhibit significant domain shift which prevents direct transfer of source split function. To compensate for this shift, we perform local normalization. If no sign correction is performed (b), we observe significant mean discrepancy which will result in erroneous model transfer between the two domains. If sign corrected (c), the mean shift between the classes significantly reduces, thus making DA feasible.

target domain prediction error, as shown below:

$$h^k_{\text{tar}} = \underset{h \in \mathcal{S}(h^k_{\text{src}})}{\text{argmin}} \; \mathbb{E}_{(\mathbf{x},\mathbf{y}) \subset (\widetilde{X^k_{\text{tar}}}, \widetilde{Y^k_{\text{tar}}})} \mathcal{P}(h(\mathbf{x}) \neq \mathbf{y}) + \mathcal{C}(h) \quad (10)$$

where $\mathcal{P}$ and $\mathcal{C}$ share definitions similar to Eq. (3).

In Eq. (10), h^k_{tar} is generated from a set of hypotheses belonging to a regularized hypothesis space $\mathcal{S}(h^k_{\text{src}})$ anchored around the source hypothesis h^k_{src}. Such an anchoring effect aids in meaningful transfer of decision boundaries between the source and target domains. Similar to the source forests, after each of the base source hypotheses is adapted, the posterior probability for an unseen target domain sample $\mathbf{x}$ is estimated as follows:

$$p_{\text{tar}}(\mathbf{y}|\mathbf{x}, X_{\text{tar}}, Y_{\text{tar}}) = \frac{1}{N} \sum_{k=1}^{N} p(\mathbf{y}|\mathbf{x}, \underbrace{h^k_t(\widetilde{X^k_{\text{tar}}}, \widetilde{Y^k_{\text{tar}}})}_{\text{adapted from } h^k_{\text{src}}}) \quad (11)$$

Each of the adapted hypothesis is parameterized by set of split nodes (with parameters θ^S_{tar} at node S_{tar}, which is adapted from node S_{src}) and leaf nodes (say, L_{tar}, which are adapted from leaf node L_{src}). Similar to the notations used in source forest training, we ignore the index k and wide tilde while referring to the target bagged dataset and the adapted hypothesis. In the subsequent subsections, we discuss the process of adapting the split and leaf nodes of each tree.

3.5.1. Adapting split node

Sign-corrected local node normalization: Following a similar approach as source domain training, we traverse down the tree adapting each split node and propagating the adaptation in a top-down fashion. Let the target domain training data that reaches a split node S_{tar} of tree k be denoted as $\mathcal{D}^S_{\text{tar}}$. The mean μ^S_{tar} and standard deviation σ^S_{tar} estimated from $\mathcal{D}^S_{\text{tar}}$ as parameters for the adapted oblique split function $\phi(\mathbf{x}, \theta^S_{\text{tar}})$. Normalizing $\mathcal{D}^S_{\text{tar}}$ during DA aids in a two-fold fashion. Firstly, it helps handle differences in dynamic ranges between variables. Secondly, it helps achieve a weak local alignment that can correct the local mean and variance shift between the domains. This effectively makes the $\mathcal{D}^S_{\text{tar}}$ comparable to its source domain counterpart $\mathcal{D}^S_{\text{src}}$, thus facilitating transfer of decision boundaries between the domains. Additionally, we ensured maximal class overlap in the normalized space by choosing the sign $sgn(i) \in \{-1, +1\}$ of $\sigma^S_{\text{tar}}(i)$ for feature i appropriately. Towards this end, we measure the class-wise mean shift between the two domains for a particular sign $sgn(i)$. It is defined as the average class-wise Euclidean distance between mean of the two domains. We choose sign that gives the minimal mean shift between

class-wise means $\mu^S_{\text{src}}(y_i)$ and $\mu^S_{\text{tar}}(y_i)$ that are the means defined for the normalized source and target data of ith feature over the samples belonging to class y which reach the split node S_{tar}. This is expressed as follows:

$$sgn(i) = \text{argmin}_{k \in \{-1, +1\}} \sqrt{\sum_{y=1}^{|\mathcal{Y}|} |\mu^S_{\text{src}}(y_i) - (k.\mu^S_{\text{tar}}(y_i))|^2} \quad (12)$$

If data of a particular class does not reach a node, it is ignored in calculation of Eq. (12). Minimizing the difference between the two domains through sign-corrected local alignment is very crucial as the distribution shift between the domains might be a mirroring shift (*i.e.* class distributions effectively on reversed with respect to the source decision boundary). Such a domain shift is illustrated in Fig. 5(a), where the local domain shift between the source and the target datasets $\mathcal{D}^S_{\text{tar}}$ is high and needs to be corrected prior to model transfer. Assuming the sign of standard deviation sgn for both the features to be positive (*i.e.* no sign correction), we reach local normalization with very high class-wise mean discrepancy $(|\mu^S_{\text{src}}(y_i) - (k.\mu^S_{\text{tar}}(y_i))|)$ between the two domains as shown in Fig. 5(b). If this is not corrected appropriately, it can lead to negative transfer, which would then trickle-down to the children split and subsequent leaf nodes in the tree. With sign correction, as shown in Fig. 5(c), the mean shift between the classes reduces significantly in comparison to Fig. 5(b), thus enabling effective transfer of the oblique split function between the domains.

Transfer relaxation: Through local node normalization, we enforce a local alignment between the source and target domains. However, the source decision boundary in the aligned space may be suboptimal for the target data and thus needs to be corrected. Towards this end, we introduce transfer relaxation as a new scheme for adapting the source split to be maximally discriminative on the target data. This is in sync with Eq. (10), where the adapted hypothesis is sought in a regularized hypothesis search subspace around the source hypothesis. In case of oblique split functions, we model this by perturbing the normalized optimal source split parameters (α^S_{tar} and α^{S0}_{tar}) as shown below:

$$\left. \begin{array}{l} \text{Coefficient: } \alpha^S_{\text{tar}} = \alpha^S_{\text{src}} + \mathcal{N}(0, \Gamma) \\ \text{Intercept: } \alpha^{S0}_{\text{tar}} = \alpha^{S0}_{\text{src}} + \mathcal{N}(0, \Gamma) \end{array} \right\} \theta^S_{\text{tar}} \in \mathcal{S}(\theta^S_{\text{src}}, \Gamma) \quad (13)$$

This randomized perturbation from the source split is controlled by the transfer relaxation parameter Γ. This hyper parameter determines the extent to which the hypothesis search space
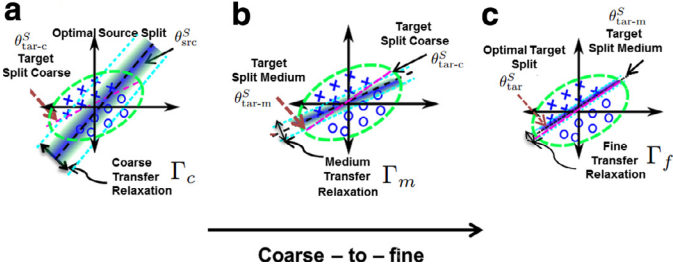
Fig. 6. Schematic illustrating hierarchical transfer relaxation to locally adapt source boundary to target domain. Firstly, the target hypothesis is sought on a coarser hypothesis search range (a) supported by source split parameters. The target split is refined in a hierarchical fashion reducing the search range and updating the support split as shown in (b, c). The gradient of the blue search space color fill is indicative of the probability of generating a candidate hypothesis in that subspace during randomized node optimization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$\mathcal{S}(\theta_{\text{src}}^S)$ around the source split is constrained. Given this and the sign corrected normalization parameters for the target data reaching node n_a^S, the oblique split $\phi(\mathbf{x}, \theta_{\text{tar}}^S)$ with parameter set $\theta_{\text{tar}}^S = \{\alpha_{\text{tar}}^S, \alpha_{\text{tar}}^{S0}, \mu_{\text{tar}}^S, \sigma_{\text{tar}}^S\}$ is defined as follows:

$$\phi(\mathbf{x}, \theta_{\text{tar}}^S) = \left(\frac{\mathbf{x} - \mu_{\text{tar}}^S}{\sigma_{\text{tar}}^S} \right) \cdot \alpha_{\text{tar}}^S - \alpha_{\text{tar}}^{S0} \tag{14}$$

The optimal set of coefficient for node S_{tar} for tree k are chosen from a family of candidate splits (say θ_c generated by randomly generating coefficients according to Eq. (13)). Each split belongs to the source regularized search space anchored around θ_{src}^S, denoted by $\mathcal{S}(\theta_{\text{src}}^S, \Gamma)$. The information gain for a particular candidate is evaluated using Eq. (6), passing candidate split at adaptation θ_c and evaluating it over $\mathcal{D}_{\text{tar}}^S$. The candidate split that gives the maximum information gain with respect to the source split is assigned the adapted node *i.e.*

$$\theta_{\text{tar}}^S = \operatorname{argmax}_{\theta_c \in \mathcal{S}(\theta_{\text{src}}^S, \Gamma)} (\mathcal{I}(\theta_c | \mathcal{D}_{\text{tar}}^S) - \mathcal{I}(\theta_{\text{src}}^S | \mathcal{D}_{\text{tar}}^S)) \tag{15}$$

The pseudo-code for this formulation is presented in Algorithm 2 in the Appendix A.

Error-correcting hierarchical formulation: In order to guide the search for the most discriminative target split (in the source-regularized search space), we deploy transfer relaxation in a hierarchical fashion. As shown in Fig. 6(a), we start with a coarser search range (Γ_c) and search for coarse target split $\theta_{S_{\text{tar-c}}}$ in $\mathcal{S}(\theta_{\text{src}}^S, \Gamma_c)$ which maximizes information gain as:

$$\theta_{\text{tar-c}}^S = \operatorname{argmax}_{\theta_c \in \mathcal{S}(\theta_{\text{src}}^S, \Gamma_c)} (\mathcal{I}(\theta_c | \mathcal{D}_{\text{tar}}^S) - \mathcal{I}(\theta_{\text{src}}^S | \mathcal{D}_{\text{tar}}^S)) \tag{16}$$

Following which, we continue refining the split by searching in the hypothesis subspace around the chosen coarse split, that is statistically constrained using medium transfer relaxation factor Γ_m. In this paper, we do not consider further hyper parameters and fix $\Gamma_m = 0.25\ \Gamma_c$ (as shown in Fig. 6(b)). The medium target split is estimated in a fashion similar to (16) replacing Γ_c by Γ_m and θ_{src}^S by the estimated coarse split $\theta_{\text{tar-c}}^S$. Further, as a final step we go to fine resolution (as shown in Fig. 6(c)), following a similar strategy and estimate the final adapted split parameters θ_{tar}^S in a fashion similar to (16) replacing Γ_c by Γ_f and θ_{src}^S by the estimated medium split $\theta_{\text{tar-m}}^S$ with $\Gamma_f = 0.25\ \Gamma_m$. The adapted split function $\phi(\mathbf{x}, \theta_{\text{tar}}^S)$ given by Eq. (14) is used to split the node-level training dataset $\mathcal{D}_{\text{tar}}^S$ into left and right subsets, using a formulation similar to Eqs. (7) and (8), respectively. This recursive adaption from the source tree continues until we reach the source leaf node or the constraints imposed by the target domain configuration parameters Θ_a (like minimum number of samples for adapting minAdapt) are satisfied or if all the target samples reaching a node belong to the same class.
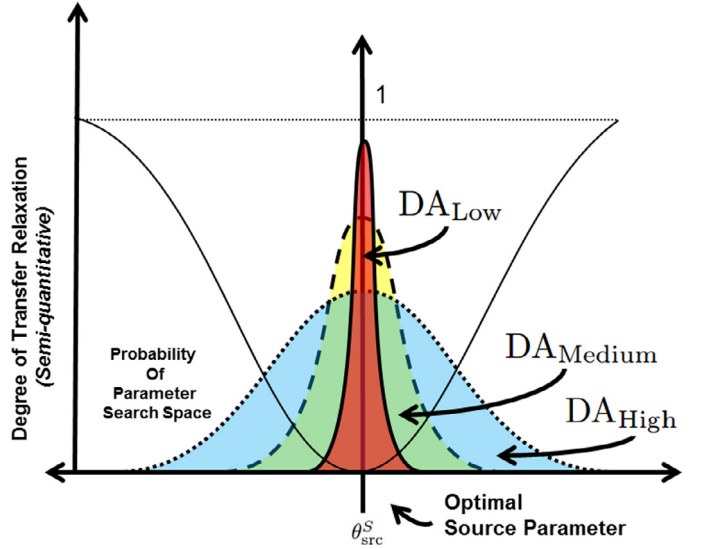
Fig. 7. Illustrative schematic showing the effect of increasing Γ with respect to bias on the source split parameters α_{src}. The variants of DA - High, Medium and Low are generated by decreasing the transfer relaxation parameter, thus increasing bias towards the source split parameters by statistically constraining the search space around them.

Effect of transfer relaxation: In this paper, we evaluate the performance of the proposed DA algorithm, varying the transfer relaxation parameter (Γ_c). These include high transfer relaxation (DA_{High}), medium transfer relaxation ($\text{DA}_{\text{Medium}}$), low transfer relaxation (DA_{Low}) and no transfer relaxation DA_{Zero}. In case of DA_{High}, with higher Γ_c as shown in Fig. 7, the search space for target split parameter is quite large, resulting in a reduced bias towards the parameters of the source split. In other words, the influence of the source split towards determining split in the target domain is low, which is quite similar to selecting coefficients through randomized node optimization solely based on target domain data. Reducing the Γ_c (from high towards low relaxation), leads to an increasing bias towards the optimal source split as a result of statistically constraining the parameter search space as shown in Fig. 7. For $\Gamma_c \to 0$, the target split is the same as the source split and it corresponds to the limiting case of *Zero* Transfer Relaxation. We hypothesize that medium relaxation would result in optimal adaptation as it balances the source-target domain bias better than the other variants of DA.

3.5.2. Adapting leaf node

Locally adaptive leaf node reweighting: Post adaptation of the parent split, let L_{tar} denote the terminal leaf node to be adapted from source leaf node L_{src}. Following the same conventions as Section 3.4.2, let $\mathcal{D}_{\text{tar}}^L$ denote the subset of adaptation data that reaches L_{tar}. The final predictive model associated with the leaf is a weighted average of the original source predictive model (denoted as $p_{\text{src}}^L(y|\mathbf{x})$) and the target domain data derived predictive model $p_{\text{tar*}}^L(y|\mathbf{x})$. The target domain data derived predictive model at this node is estimated (similar to the source leaf predictive model in Eq. (9)) as:

$$p_{\text{tar*}}^L(y|\mathbf{x}) = \frac{|\mathcal{D}_{\text{tar}}^L(\mathbf{y} = y)|}{|\mathcal{D}_{\text{tar}}^L|} \tag{17}$$

Given this, the leaf node model p_{tar}^L is estimated as:

$$p_{\text{tar}}^L(y|\mathbf{x}) = (1 - \gamma)\ p_{\text{tar*}}^L(y|\mathbf{x}) + \gamma\ p_{\text{src}}^L(y|\mathbf{x}) \tag{18}$$

where γ is a free scalar parameter that weights the influence of the source against the target domain. In this paper, we fix γ as

follows:

$$\gamma = \frac{(|\mathcal{D}_{\mathrm{src}}^{L}|/|\mathcal{D}_{\mathrm{src}}|)}{(|\mathcal{D}_{\mathrm{tar}}^{L}|/|\mathcal{D}_{\mathrm{tar}}|) + (|\mathcal{D}_{\mathrm{src}}^{L}|/|\mathcal{D}_{\mathrm{src}}|)} \qquad (19)$$

The weighting factor is locally-determined from the target and source training data that reach that leaf node and hence coined as locally adaptive. This term in Eqs. (18) and (19) creates allots higher weight towards the domain which has a higher fraction of data reaching the node. Taking fractions effectively cancels out the possible additional bias created due to the limited availability of samples from target domain in comparison to the source domain.

Split to leaf node conversion: As the target domain dataset for adaptation is relatively smaller than the source domain dataset, it is possible that certain branches in the source tree may not be traversed, and hence not adapted. This arises if the target feature space does not entirely span the source feature space parsed by these branches of the source tree. We prune these branches by converting the child split node of the bottom-most traversed split node into a leaf node. Mathematically, let $S \rightarrow L_{\mathrm{tar}}$ be leaf node to be generated from source split node S_{src} and the family of leaf nodes in branches grown from this node be denoted as $\mathcal{FL}(S_{\mathrm{src}})$. The predictive model associated with the new node $p_{\mathrm{tar}}^{S \rightarrow L}(y|\mathbf{x})$ is estimated as a weighted average of all the predictive models of leaves in $\mathcal{L}(S_{\mathrm{src}})$ as follows:

$$p_{\mathrm{tar}}^{S \rightarrow L}(\mathbf{y}|\mathbf{x}) = \frac{1}{|\mathcal{D}_{\mathcal{FL}(S_{\mathrm{src}})}|} \sum_{l \in \mathcal{FL}(S_{\mathrm{src}})} |\mathcal{D}_{l}| \times p_{l}(\mathbf{y}|\mathbf{x} \in \mathcal{R}_{l}) \qquad (20)$$

It must, however, be noted that, if the adapting dataset is fully representative of the target feature space, no new unseen sample would traverse down to the converted split node and hence this exclusively source derived predictive model has no effect whatsoever (effectively absent). In cases otherwise, we use the predictive model of Eq. (20) as an approximation.

Scenario when $|\mathcal{D}_{\mathrm{tar}}^{S}| < \mathtt{minAdapt}$: To prevent over-fitting while adapting at each split node, we impose a termination criterion of minimum number of target labeled samples required for adaptation ($\mathtt{minAdapt}$). In such a scenario when the number of target domain samples reaching split node S ($|\mathcal{D}_{\mathrm{tar}}^{S}|$) is less than $\mathtt{minAdapt}$, the corresponding split node S_{src} of source tree is converted into a leaf node $S \rightarrow L_{\mathrm{src}}$ in a fashion similar to Eq. (20) using the family of source leaf nodes grown from this split node ($\mathcal{FL}(S_{\mathrm{src}})$). Let the predictive model associated with the generated leaf node be denoted as $p_{\mathrm{src}}^{S \rightarrow L}(\mathbf{y}|\mathbf{x})$ with the total number of source domain samples reaching this split node as $|\mathcal{D}_{\mathcal{FL}(S_{\mathrm{src}})}|$. Following this conversion, the leaf node is adapted in the same fashion as any other leaf node using the locally adaptive leaf node reweighting approached using Eqs. (17)–(19).In this scenario, the target derived model used in (17) is derived from $\mathcal{D}_{\mathrm{tar}}^{S}$; $p_{\mathrm{src}}^{S \rightarrow L}(\mathbf{y}|\mathbf{x})$ and $|\mathcal{D}_{\mathcal{FL}(S_{\mathrm{src}})}|$, are used instead of $p_{\mathrm{src}}^{L}(y|\mathbf{x})$ and $|\mathcal{D}_{\mathrm{src}}|$ respectively in Eqs. (17)–(19).

3.6. Baselines

In addition to domain adaptation as a solution for *in vivo* TC, we explore other plausible solutions as baselines for comparative analysis *vs.* the proposed method. These include direct deployment of *in vitro* trained forest on aligned *in vivo* data (BL1), training forest only with the limited *in vivo* dataset (BL2) and lastly jointly training with the *in vitro* and *in vivo* data to generate a domain-invariant classifier (BL3). BL3 is conceptually similar to transfer learning in decision forests (Goussies et al., 2014). The hypothesized ideal performance curves for the baselines and the proposed method is illustrated in Fig. 8. In this context, we hypothesize that our proposed approach of domain adaptation of random forests with appropriate choice of transfer relaxation (DA with TR) produces better learning curves characterized by a higher performance
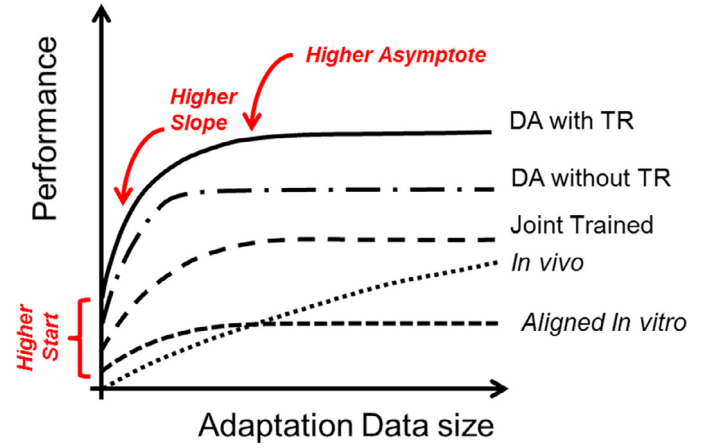

Fig. 8. Schematic demonstrating three metrics to evaluate domain adaptation performance, with increasing size of the adaptation dataset. (Figure adapted and suitably modified from Tommasi et al., 2014.)

start, greater slope, and higher asymptote than the other closely related baselines as illustrated in Fig. 8.

4. Experiments and discussions

4.1. Data description and configuration settings

We validate our proposed method in characterizing heterogeneous atherosclerotic tissues on IVUS, where direct deployment of *in vitro* trained classifier for *in vivo* application is challenging due to the presence of blood which induces a domain distribution shift. The *in vitro* radio frequency ultrasound data used in these experiments is acquired from 53 coronary artery cross-sections from post-mortem or transplantation of 13 human hearts (Katouzian et al., 2007), circulating saline in the lumen using commercially available [2] single-element 40 MHz catheter. The radio frequency data was acquired in the polar domain with each cross-section of 2048×256 at a sampling frequency of 400 MHz. Subsequently, conventional histology (Hematoxylin and Eosin and Movat Pentachrome) was obtained for validation. (For detailed description of the protocol, refer to Katouzian and Laine (2010)).

The *in vivo* data is available for seven artery cross-sections from 3 human hearts. We obtained near-realistic *in vivo* setting by circulating human blood maintaining arterial pressure and pulsatile flow. This approach is observed to closely mimic IVUS acquisition for *in vivo* settings as corroborated by an expert interventional cardiologist. The cost and complexity involved and strict inclusion criteria in terms of the disease stage, availability of histology and human blood made *in vivo* data collection extensively challenging.

To remove any bias that could be potentially introduced in training/adaptation due to a possibly close by match between the *in vitro* and *in vivo* datasets from the 7 common arteries, we completely separate the 53 *in vitro* samples into the *in vitro* training dataset comprising of 46 samples and the matched dataset comprising of the 7 cross-sections. We match the *in vitro* data from these 7 artery cross-sections to their corresponding *in vivo* data and utilize it to contrast the predicted tissue posterior probabilities (shown in pseudo-color in Fig. 10). The ground truth for the application is derived from matched *in vitro* TC results derived from our state of the art *in vitro* TC framework (Stochastic Driven Histology (SDH) (Sheet et al., 2014)). These labels were correlated with the corresponding histology slice and corrected by an expert interventional cardiology using ImageJ (Schneider et al., 2012). Histological

---

[2] Atlantis IVUS Catheter, Boston Scientific, Fremont, CA, USA.

Table 2
Parameter settings for the baselines and proposed method (all includes BL1, BL2, BL3 and all variants of DA).

Parameters	Settings	Applicable to
Subspace dimension (d)	30	All
Depth of the tree(treeDepth)	∞	(for DA, it is retained from source forest)
Number of trees (numTrees)	5, 10, ···, 50	
Bagging ratio (bagRatio)	5 %	
Min. samples to split (minSplit)	50	
No. of candidate splits (numSplit)	50	
No. of variables in split (featSplit)	2	
Min. samples to adapt (minAdapt)	10	All DA
Transfer relaxation (Γ_c)	0.5	DA$_{High}$
	0.1	DA$_{Medium}$
	0.01	DA$_{Low}$
	0	DA$_{Zero}$

validation revealed that plaques exhibit co-located heterogeneity (which is considered in SDH). Manual tracings on IVUS images of tissues generally mark large regions to be belonging to one tissue type, which is contrary to what is observed on histology. As such, matched results from SDH are more reliable representatives of tissue heterogeneity in contrast to manual labels. Additionally, since the datasets are matched, the tissue composition should be invariant of the transmission medium in the lumen (saline for in vitro and blood for in vivo), which further justifies use of matched *in vitro* TC observations as ground truth. The *in vivo* and the matched *in vitro* artery cross-sections are registered using control-point registration with thin plate splines deformation model. The inferred deformation field is used to transform probabilistic labeling derived from *in vitro* TC to align with the corresponding *in vivo* artery cross-section. The maximum a posteriori class of tissues is used as the discrete label for adaptation and evaluation. These labels were then transformed to the polar coordinate space of *in vivo* IVUS RF data for creating learning examples for this work.

Configuration settings: The parameter configurations for the baselines and the proposed method are tabulated in Table 2. In this application, we utilize bivariate splits at each split node. While choosing higher order splits (featSplit >2), the number of free-parameters grows in proportion to the chosen order. In addition to higher training/adaptation computational time incurred, if the order is chosen very high, the forest might over-fit during randomized node optimization while training and adaptation, due to limited number of target data available. Specific to the application at hand, we observed that forest with configuration featSplit = 2, performed better than univariate and other higher order splits and is therefore used in all subsequent validations.

The algorithms were implemented using MATLAB R2014b on workstation with 64GB RAM memory and 2.67 GHz Intel Xeon (R) processor. It is important to note that, DA is controlled by one free parameter only, the transfer relaxation Γ_c, and the rest of the configuration is retained from the source forest. In an typical setting, we assume that the source forest is reliably trained where the choice of configuration parameters is optimal and well validated such that a source forest is readily available for adaptation to a target domain.

4.2. Time complexity analysis

The worst-case time-complexity of the domain adaption of the source forest is $\Omega(MKL\widetilde{N}\log^2 \widetilde{N})$, where M is numTrees to be adapted and N number of samples in the adapting dataset ($\widetilde{N} =$ bagRatio $\times N$, due to bagging), K is number of splits evaluated at each leaf node and L is the number of levels in the hierarchical transfer relaxation (here, $L = 3$). $\log^2 \widetilde{N}$ is the worst case number

of nodes for extremely deep trees with minSplit= 1. This is independent of the value of chosen transfer relaxation. For the other baselines (BL1, BL2, and BL3), the worst-case time complexity is the same as above, except that $K = 1$ as only randomized node optimization is employed.

4.3. Evaluation metrics

Visual evaluation: In vivo TC falls under the premise of multi-class probabilistic learning due to presence of heterogeneous tissues. For visual evaluation, we retain the probabilistic nature of the estimates and generate pseudo-colored tissue probability maps as shown in Fig. 10. Each constituent tissue is color-coded: calcified (blue), lipidic (yellow), as well as fibrotic (pink) and the predicted posterior probabilities are alpha-blended to generate this representation. Contrasting with matched *in vitro* dataset aids in very effective visual comparison of the performance of the baselines. If the assumption that the source classifier is reliably trained holds, we hypothesize that upon adapting from it, the tissues predicted *in vivo* must be visually similar to their matched *in vivo* counterparts as the tissue compositions do not change between the domains.

Quantitative evaluation: For quantitative evaluation, we use ground truth obtained by taking maximum *a posteriori* of the probabilities derived from matched *in vitro* TC results and corrected by an expert interventional cardiologist in corroboration with the observed histology. The prediction performance is evaluated only within the plaque area.[3] We use single-test indicators like log-compressed multi class diagnostic odds ratio (logDOR) in addition to conventional metrics like area under receiver operating characteristics curve (AUC) and average prediction accuracy (Acc) to assess the performance of the competing methods. The logDOR is calculated as follows:

$$\text{logDOR} = \frac{1}{|\mathcal{Y}|} \sum_{\forall y \in \mathcal{Y}} \log_e \left(\frac{TP(y)/FN(y)}{FP(y)/TN(y)} \right) \tag{21}$$

where for class *y, TP, FN, FP* and *TN* refers to true positive, false negative, false positive, and true negative from class-wise confusion matrix generated through a *one vs. rest* binarization of tissue labels (Glas et al., 2003). To assess TC from a segmentation perspective, we also calculate the Sørensen–Dice coefficient for each tissue type. Unless specified, to make each evaluation metric statistically meaningful, we calculate it as an average of 10 experimental runs which differ from one another due to the randomization involved in forest training.

Adaptation performance: It is interesting to understand how the learning curve progresses with increasing availability of samples for adaptation/training of *in vivo* classifiers. This is indirectly assessed under *k-folded* cross-validation increasing *k* from 1 to *leave-one-out*.

We designed a set of experiments to assess the comparative performance of the proposed DA method over the baselines in terms of its overall performance, sensitivity to number of base learners, and domain adaptation performance evaluating the metrics discussed earlier in the section.

4.4. Training in vitro source forest

In this experiment, we divide the 46 *in vitro* training datasets into 5 strictly-disjoint folds of 9 datasets each (one subset with 10 datasets), which were further subdivided into two disjoint training subsets each, thus resulting in 5 × 2 data subsets. This is done

---

[3] We manually segment the lumen border and the media adventitia borders *a priori* to demarcate the plaque area. However, this process can also be automated using algorithms presented in Katouzian et al. (2012b).

Table 3
Folded cross validation testing efficacy for selecting *in vitro* data fold for adaptation.

	Fold 1.1	Fold 1.2	Fold 2.1	Fold 2.2	Fold 3.1	Fold 3.2	Fold 4.1	Fold 4.2	Fold 5.1	Fold 5.2
logDOR Performance	4.15 ± 0.28 Worst	4.15 ± 0.29	5.24 ± 0.22	5.25 ± 0.22 Medium	5.03 ± 0.21	5.01 ± 0.22	5.12 ± 0.21	5.09 ± 0.21	5.74 ± 0.19 Best	5.72 ± 0.18

Table 4
Domain differences and TC performance before and after domain alignment measured using Subspace Disagreement Measure (SDM) (Gong et al., 2012) and logDOR metric.

K-fold	(Prior to alignment)		(Post-alignment)		p-Value
	ΔD	logDOR	ΔD	logDOR	
$k = 1$	13.12 (±0.04)	2.85 (±0.20)	7.26 (±0.71)	3.68 (±0.16)	$< 10E - 05$
$k = 2$	14.09 (±0.15)	2.84 (±0.21)	7.01 (±1.37)	3.68 (±0.15)	$< 10E - 05$
$k = 3$	16.11 (±0.04)	2.86 (±0.33)	6.54 (±1.08)	3.51 (±0.23)	$< 10E - 05$
$k = 4$	14.57 (±0.26)	2.87 (±0.39)	6.54 (±1.03)	3.64 (±0.43)	$< 10E - 05$
$k = 5$	14.09 (±0.02)	2.78 (±0.38)	6.16 (±0.79)	3.57 (±0.60)	$< 10E - 05$
$k = 6$	15.08 (±0.73)	2.73 (±0.40)	5.87 (±0.714)	4.29 (±0.48)	$< 10E - 05$

due to the computational limitation to training data size imposed by the available heap memory in the used computational platform. Further, the disjoint folds belonging to the same dataset they act as training and validation sets interchangeably. Individual source forests were trained using the configuration parameters described in Table 2 with the source domain training method discussed in Section 3.4. Due to randomization involved, forests were independently generated 10 times for each training subset (say Fold X.1) and the forest that performed the best on the corresponding validation dataset (Fold X.2) is chosen as the optimal model representing that fold (Fold X.1). This is repeated interchanging the roles of the two subsets and an optimal model representing Fold X.2 is chosen by validating on Fold X.1. We evaluate the testing efficacy of selected representative model of particular data subset using *one vs. rest* cross-validation by testing on the rest of strictly-disjoint folds. Table 3 demonstrates the average and error bars of the logDOR values obtained during the *in vitro* testing. To analyze the sensitivity of adaptation performance to the choice of *in vitro* source model (in Section 4.5.3), we choose the representative models from three differently performing source folds (Fold 5.1, Fold 2.2 and Fold 1.1 corresponding to best, medium and worst source model in terms of testing efficacy) for further validations.

4.5. Domain adaptation validation experiments

We explore the ability of the baselines and the variants of the proposed model to learn with minimal samples. To avoid any bias, we use *k*-folded cross-validation, dividing the *in vivo* data artery-wise into *k* target training/adaptation dataset and $(7 - k)$ testing datasets. Varying *k* from $k = 1$ to *leave-one-out* $(k = 6)$ we generate 10 train/test dataset dyads (except for $k = 6$ where we could create only 7 dataset dyads).

4.5.1. Evaluation of domain difference after domain alignment

Aligning domains will help compensate for the global inter-domain distribution shift, thus making the decision models trained on the aligned source space comparable over the source-aligned target domain. As hypothesized earlier, the difference between the domains (ΔD) would be minimized upon alignment. As we use an unsupervised domain alignment formulation through rPCA, it is befitting to measure domain difference (ΔD) as a measure of mis-alignment between the basis vectors of the two domains (say $\mathcal{P}_{src}$ and $\mathcal{P}_{tar}$). Towards this, we use the subspace disagreement measure presented in Gong et al. (2012) as a measure for domain difference. We combine both domain datasets into one and estimate

its basis vectors (say $\mathcal{P}_{src+tar}$). Intuitively, if the two domains were aligned, the three subspaces would not be far apart on the Grassmanian manifold (Hamm and Lee, 2008). As reported in Gong et al. (2012), ΔD captures this in terms of the principle angles between the subspaces. Let η_i denote angle between *i*th basis vector of $\mathcal{P}_{src}$ and $\mathcal{P}_{src+tar}$ and ξ_i between $\mathcal{P}_{tar}$ and $\mathcal{P}_{src+tar}$. These are measured as:

$$\eta_i = \arccos\left(\frac{\langle \mathcal{P}_{src}(i).\mathcal{P}_{src+tar}(i)\rangle}{\|\mathcal{P}_{src}(i)\|\|\mathcal{P}_{src+tar}(i)\|}\right) \tag{22}$$

and

$$\xi_i = \arccos\left(\frac{\langle \mathcal{P}_{tar}(i).\mathcal{P}_{src+tar}(i)\rangle}{\|\mathcal{P}_{tar}(i)\|\|\mathcal{P}_{src+tar}(i)\|}\right) \tag{23}$$

The subspace disagreement measure for the *i*th feature SDM(i) is given by:

$$SDM(i) = 0.5[\sin \eta_i + \sin \xi_i] \tag{24}$$

For the optimal subspace dimension d, the domain difference is calculated as $\Delta D = \sum_{i=1}^{d} SDM(i)$. For perfectly aligned basis vectors, $\eta_i = 0$ and $\xi_i = 0$; therefore SDM(i) = 0. So, the lower the ΔD is, better the domains are aligned. The upper bound of the lower-dimensional space for projecting and aligning the two domains was estimated as $d = 30$ using method discussed in Fernando et al. (2013). We further observed that at $d = 30$, $>99\%$ of the data variance is preserved, indicating minimal loss in information due to dimensionality reduction

To validate the domain alignment hypothesis, we compared the domain difference measures prior and post alignment calculated over each of the data-folds in the *k* fold cross-validation and tabulated the results in Table 4. We perform non-parametric Two-sample Kolmogorov–Smirnov statistical test to evaluate the statistical significance of domain alignment. In addition to measuring SDM, we also evaluate the performance of TC prior and post subspace alignment. For prior to alignment setting, the logDOR metric is calculated by directly deploying model trained on source domain to the target domain data. For post alignment setting, the logDOR metric is calculated by deploying model trained on target-aligned source data to the target domain test data (corresponds to baseline BL1).

Observations: From Table 4, we observe that domain alignment of the subspaces significantly reduces the domain differences between the source and target domains (p-Value $< 10E - 05$ implying statistically significant results). We also observe that as *k* increases, the domain differences decrease given the availability of larger and
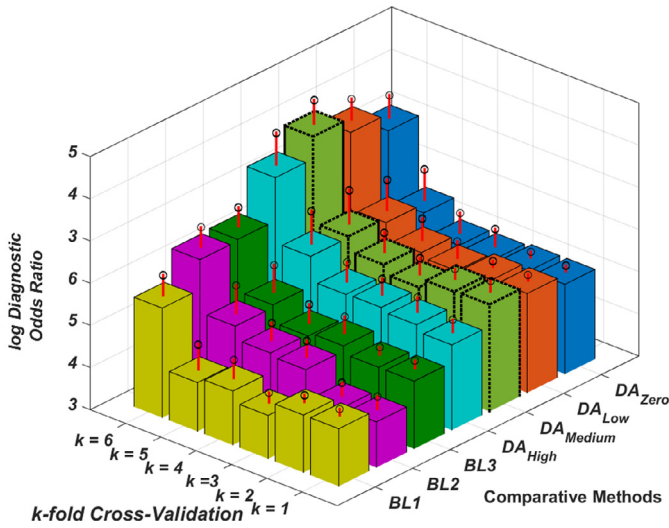
Fig. 9. *In vivo* folded cross validation testing results for fixed `numTrees = 50`. The average testing error with 95% confidence limits evaluated using logDOR metric is plotted for the different comparative methods.

hence better representative datasets for alignment. Analyzing TC performance prior to alignment, we observe that the logDOR metric shows no statistically significant difference as *k* is increased. This is expected as this is a scenario of no adaptation and no target domain samples are used. Post alignment, we observe a significant improvement in logDOR metric in comparison to prior alignment, indicating that compensating global domain shift through alignment makes the source model trained on aligned data more suitable for target domain over its direct deployment.

### 4.5.2. Influence of adaptation dataset size

Upon alignment, the source forest trained with *in vitro* data is adapted through the proposed DA method for each of the DA

variant considering DA_{High}, DA_{Medium}, DA_{Low} and DA_{Zero}. In addition, the classifiers for the baselines BL1 (forest trained on aligned source data), BL2 (forest trained on target data) and BL3 (domain invariant forest trained with combined source and target data) are also generated. The parameters used for each of these methods are tabulated in Table 2, with the `numTrees` fixed at 50 for this experiment. Following this, the trained/adapted classifiers are tested on the strictly-separated *in vivo* testing dataset of that particular dataset dyad. We calculate the average logDOR and the 95% confidence limits of the testing performance for the different *k* values and comparative methods. The results are illustrated in Fig. 9. Visual example of three test *in vivo* artery cross-sections along with matched *in vivo* cross-sections and corresponding hematoxylin and eosin histology is shown in Fig. 10.

Observations: All comparative methods monotonically improve with increasing number of training/adapting exemplars. We observe a consistent trend that all the variants of DA, improve significantly over the baselines. This observation supports our hypothesis that leveraging *in vitro* data improves TC performance *in vivo*. Despite correcting for global domain shift through domain alignment, direct deployment of *in vitro* classifier is sub-optimal on the aligned *in vivo* data, owing to local domain shifts that have not been corrected. We observe that BL3 which is a domain-invariant classifier is superior to both BL1 and BL2, implying that *in vivo* is usable for *in vitro* classification; however, the model has to be made maximally discriminative on the target domain. Additionally, a visual evaluation on matched *in vivo–in vitro* artery cross-sections in Fig. 10 demonstrates that the proposed method predicts the confluent tissue pools with maximum similarity to the tissue probabilities predicted by the *in vitro* classifier on *in vitro*, especially the calcification arc (blue-colored tissue) and confluent tissue pools surrounding it. In Fig. 10(a) and (c), we have indicated with yellow arrows regions behind the arc of calcification where the ultrasonic information carries very low confidence (due to high backscattering by hard calcium). Both the *in vitro* and the proposed DA *in vivo* TC results classify this region as non-tissue (dark in the pseudo-color rendering Fig. 10(b) and (g), respectively), which is
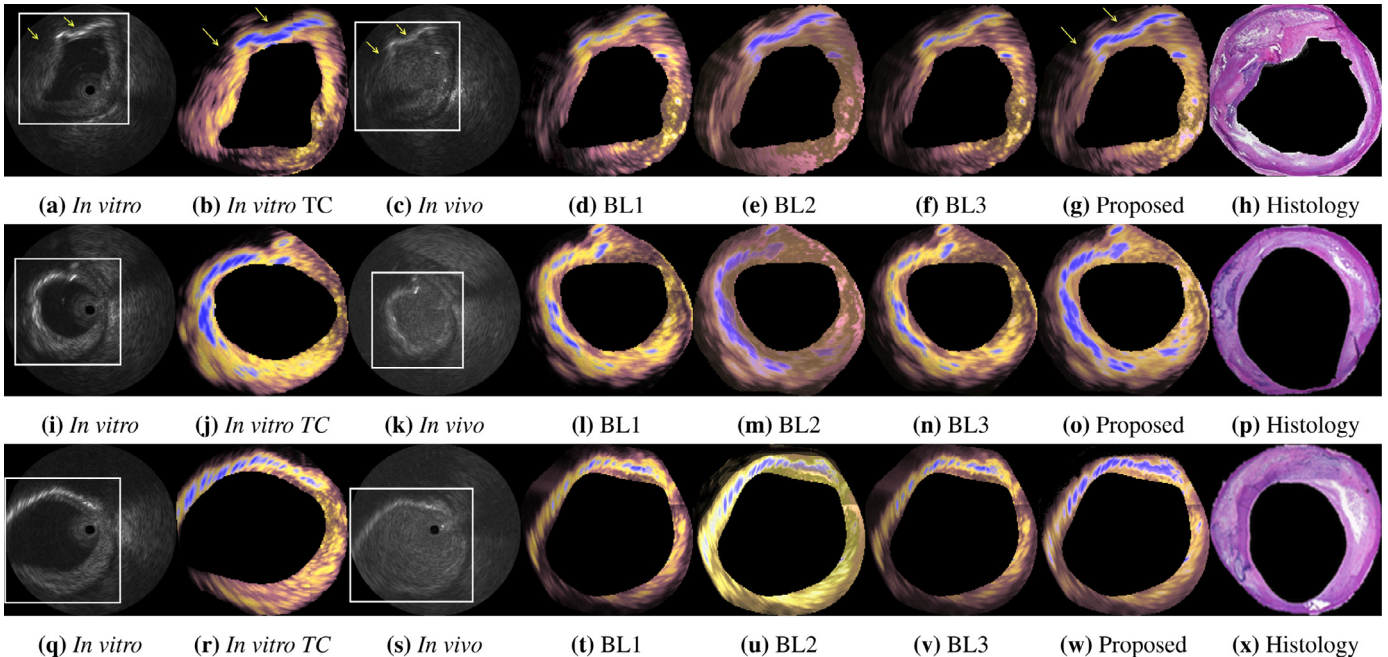
Fig. 10. Illustration of the source domain (*in vitro*) image (a, , q) with corresponding target domain (*in vivo*) image (c, k, s) and conventional histology (h, p, x). The results using the proposed method (g, o, w) and baselines (BL1 – (d, l, t); BL2 – (e, m, u) and BL3 – (f, n, v) are rendered in pseudo-color (*blue* – calcified, *pink* – fibrotic and *yellow* – lipidic). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
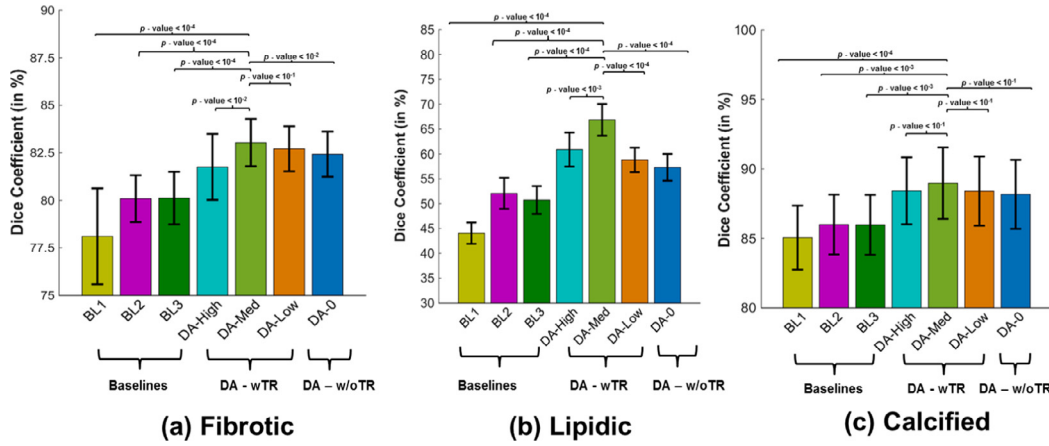
Fig. 11. Tissue-wise domain adaptation segmentation performance (Sørensen Dice coefficient) of proposed method (DA − wTR) *vs.* baselines (DA − w/o TR and BL1-3) for (a) fibrotic, (b) lipidic and (c) calcified tissues. In addition, we also report the *p*-Values evaluated using non-parametric Two-sample Kolmogorov–Smirnov statistical test between performance of DA_{Medium} *vs.* the other methods.

Table 5
In vivo testing results (logDOR metric) for different source folds and effect of hierarchical transfer relaxation ($k = 4$ cross-validation with `numTrees`= 50.)

Source fold	Baselines			Variants of DA				
	BL1	BL2	BL3	Hierarchy?	DA_{High}	DA_{Medium}	DA_{Low}	DA_{Zero}
Fold 1.1	3.08 ± 0.33	3.74 ± 0.36	3.72 ± 0.34	w/o	3.93 ± 0.38	3.94 ± 0.41	3.80 ± 0.39	3.79 ± 0.38
				With	4.06 ± 0.42	**4.12 ± 0.42**	3.94 ± 0.43	
Fold 2.2	3.47 ± 0.30	3.74 ± 0.36	3.79 ± 0.37	w/o	4.02 ± 0.39	4.05 ± 0.43	4.03 ± 0.47	4.00 ± 0.44
				With	4.17 ± 0.44	**4.18 ± 0.48**	4.16 ± 0.47	
Fold 5.1	3.64 ± 0.44	3.74 ± 0.36	3.84 ± 0.45	w/o	3.97 ± 0.44	4.06 ± 0.46	4.04 ± 0.47	4.03 ± 0.47
				With	4.12 ± 0.46	**4.25 ± 0.46**	4.17 ± 0.51	

Note: The best performance for each *k*-fold is shown in boldface and the best result amongst all folds is frameboxed . The mean and standard deviation of logDOR metric are tabulated for the baselines and variants of proposed DA algorithm.

justified as tissue class cannot be inferred with confidence from backscattered signals received from such shadow regions (Tanaka et al., 2007).

Comparing variants of DA, we observe a consistent trend of $DA_{Medium} > DA_{Low} > DA_{Zero} > DA_{High}$ across all *k* values. This observation is consistent with our hypothesis that a medium transfer relaxation leads to better tradeoff between source and target domain biases. Further, the effect of transfer relaxation as discussed earlier in Section 3.5.1 is also empirically supported by our observations. We also evaluate the tissue-wise segmentation performance of the proposed variants of DA and the baselines using Sørensen–Dice coefficient and the results for $k = 4$ cross-validation are reported in Fig. 11. We observe that the variants of DA outperform the baselines for all the tissue types. In comparison, the margin of improvement for segmentation of Lipidic classes are significantly higher for DA_{Medium} over other variants ($>15\%$ in comparison to BL3). In case of calcified class, the detection performance across the variants of DA and baselines is not significantly different ($<5\%$ in comparison to BL3). This can also be attributed to well-known observation due to strong and distinguishing hyper-echoic ultrasound backscattering signatures associated with hard calcium in comparison to other soft tissue constituents of the plaque (lipid and fibrous tissues), which makes its detection relatively easier. Observations based on Sørensen–Dice coefficient are consistent with logDOR scores.

4.5.3. Evaluation of choice of in vitro source model and effect of fine-tuning with hierarchical transfer relaxation

The proposed framework for domain adaptation of random forests falls under the purview of model transfer methods for domain adaptation, wherein the source domain model parameters are suitably modified to be maximally discriminative on a related target domain. We hypothesize that a reliable source model would lead to better adaptation in the target domain. It is also interesting to study the effect of starting with a lesser reliable source model and validate if the proposed adaptation approach is able to successfully augment and adapt its parameters for the target domain. Towards this end, we choose three differently performing *in vitro* source models (Best performance: Fold 5.1; Mid-level performance: Fold 2.2 and Worst performance: Fold 1.1 as reported in Table 3). We perform $k = 4$ folded cross-validation on the target domain and tabulate the mean and standard deviations of logDOR metric observed for varying source folds in Table 5. In addition to validating the effect of the choice of source model, we also investigate the improvement observed due to introducing hierarchical scheme during transfer relaxation. The coarse, medium and fine relaxation stages of transfer relaxation are to be seen as fine-tuning stages. We validate if the proposed fine-tuning improves results across all variants of DA by contrasting the results obtained if only coarse transfer relaxation is performed. These results for $k = 4$ fold cross-validation are presented together with the results of varying source folds in Table 5.

Observations: From an overall perspective, for all three fold combinations we observe that DA_{Medium} significantly outperforms other variants of DA and the baselines. Additionally, we observe that the performance of variants of DA and baselines BL1, BL3 strictly follows the ranking of Fold 5.1 > Fold 2.2 > Fold 1.1 in terms of TC performance. This observation corroborates our hypothesis that the more effective the model in the source domain, better the model transfer performance through the proposed adaptation technique. We also observe that the performance improvement induced by adaptation is consistent across all the chosen
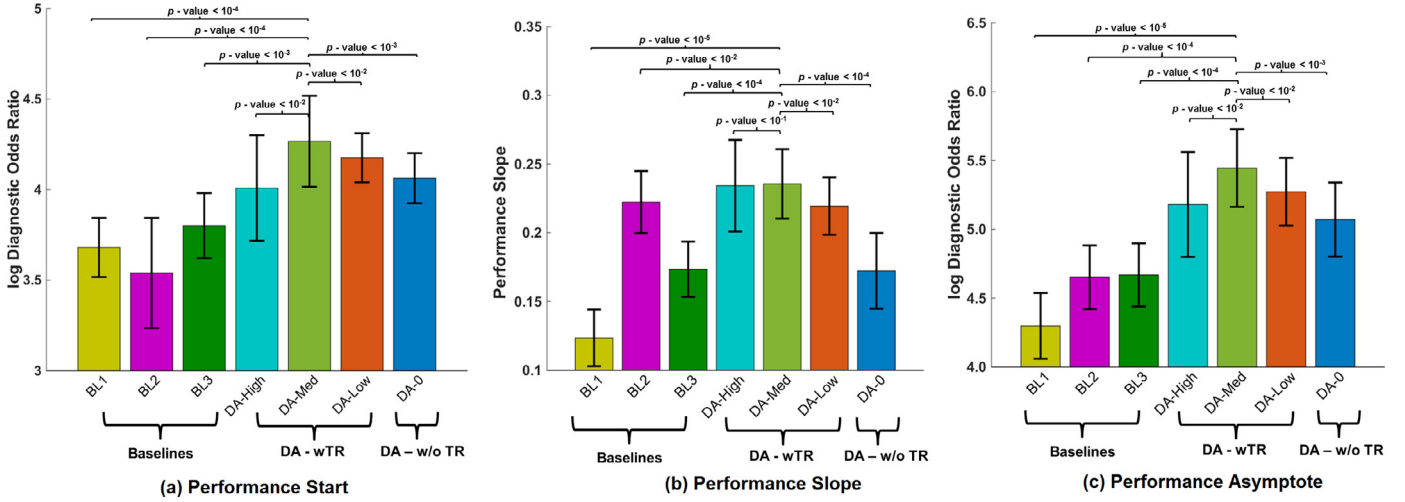
Fig. 12. Experiment 3: domain adaptation performance of proposed method (DA – wTR) *vs.* baselines (DA – w/o TR and BL1-3) with reference to Fig. 8. (a) Performance start, (b) performance slope and (c) performance asymptote. In addition, we also report the *p*-Values evaluated using non-parametric Two-sample Kolmogorov–Smirnov statistical test between performance of DA_{Medium} *vs.* the other methods.

folds, thus demonstrating that adaptation is effective even if the source model is sub-optimal. Additionally, to evaluate the effect of hierarchy in the transfer relaxation, we observe significant improvement for all variants of DA and across different source folds. This improvement is owed to the fine-tuning of the source split parameters which makes the constrained search about the source split parameters more principled. It must also be noted that if no fine-tuning is made, the DA variants still perform significantly better than the baselines. This observation validates our claim that transfer relaxation is an effective adaptation strategy and fine-tuning with the proposed hierarchical scheme helps.

4.5.4. Evaluation from knowledge transfer perspective

We calculate the improvement rate with addition of exemplars (increasing k) by fitting a linear function to the performance curve and estimating its slope, denoted as the Performance Slope. Paying attention to the cases of *one vs. rest* cross validation ($k = 1$) and *leave out one* cross-validation ($k = 6$), which correspond to Performance Start and Performance Asymptote as illustrated in Fig. 8. It must be noted that ideal asymptotic convergence of the learning performance curves as shown in Fig. 8 is not achieved for any of the baselines and DA variants used, so the results of $k = 6$ is used instead. The results observed on the comparative methods are plotted in Fig. 12.

Observations: This experiment is performed to validate the hypothesis proposed in Fig. 8. Taking a closer look at Fig. 12, we observe that the proposed method with medium transfer relaxation DA_{Medium} has significantly higher performance start and performance asymptote over the comparative methods. (In Fig. 12, the statistical significance estimated using non-parametric Two-sample Kolmogorov–Smirnov statistical test and resultant *p*-Values are reported.) In terms of performance slope, our approach DA_{Medium} and baselines BL2 and DA_{High} have comparative performance and our approach has significantly higher than the other methods.

4.5.5. Parameter sensitivity to numTrees

As reported by Breiman (2001), performance of ensemble methods like random forests is directly a result of the bias-variance tradeoff. In forests, increasing number of trees reduces the variance and improves its generalization performance. In this context, it is interesting to evaluate if adaptation through the proposed approach improves the strength of the trees. Ideally, if the model

transfer is successful between the source and the target domain, there will be a monotonic increase in performance until sufficient number of trees are added to the ensemble, beyond which the performance will converge. If adaptation reduces the strength of the trees (called as negative transfer), the performance will degrade or saturate early with increasing numTrees. We validated this for the comparative methods evaluating the performance varying numTrees from 5 to 50 in increments of 5 trees using AUC, Acc, and logDOR as metrics. We used 10 randomly generated strictly-separated adapting – testing *in vivo* datasets using $k = 4$ cross validation and report the average performance with the 95% confidence interval error bars in Fig. 13. In addition to evaluation of the parameter sensitivity to numTrees, we also report the average training time per tree (τ_{tar}) for each of the baselines and the proposed method.

Observations: With respect to overall performance across all metrics, DA_{Medium} is consistently superior than the comparative methods. In addition, we notice that the gap between DA_{Medium} and DA_{Low} is decreasing as numTrees increases. This shows that the effect of optimal transfer relaxation is most profound for smaller number of trees. It also shows that asymptotically (for sufficiently many trees), fine-tuning of the transfer relaxation parameter is not necessary as DA achieves similar results for a wide range of Γ_c values. Further, with respect to improvement in the strength of the forest with increasing numTrees, we observe that DA_{Medium}, DA_{Low} and DA_{Zero} present an overall monotonic trend in performance improvement across different metrics. The baselines of BL2 and BL3 also display the expected increasing trend as they are trained independently in each run and do not involve any model adaptation. Also, we observe that DA_{High} shows a non-monotonic trend beyond 20 trees, indicating a possible scenario of negative knowledge transfer. Since forests are adapted tree-wise in a top-down fashion, high deviation from source boundary at the top nodes of the tree might render the source-forest splits ineffectual and the error due to this would propagate down the tree leading to negative transfer. BL1 exhibits a weakly monotonic change as numTrees increases but converges soon after 20 trees. The increasingly closer gap between DA_{High} and BL2 as numTrees increases supports our hypothesis that very high transfer relaxation effectively removes any bias induced by the source-split's anchoring effect and is very close to random node optimization that is used for training BL2.
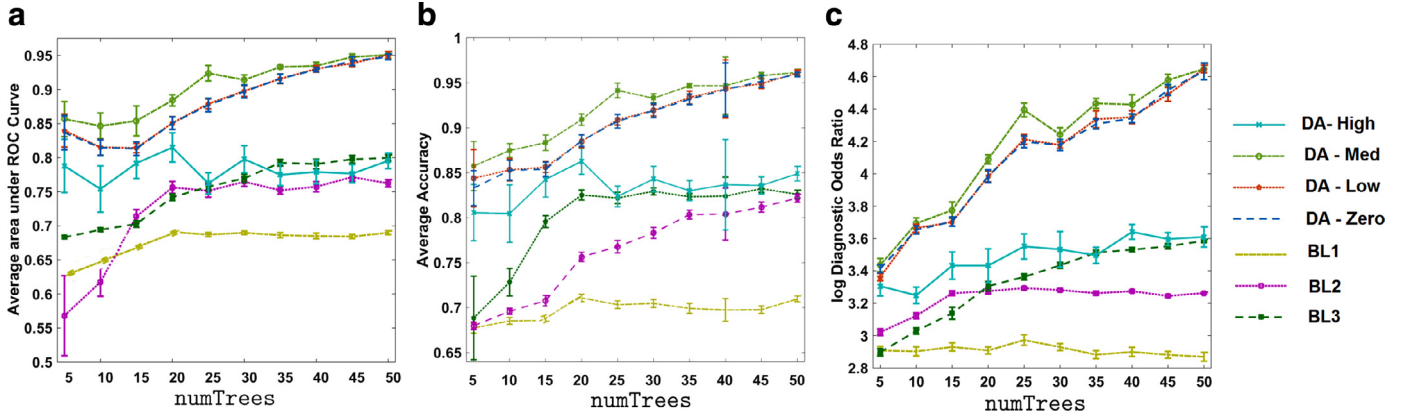
Fig. 13. Evaluating the strength of the forest varying `numTrees` for $k = 4$ folded cross validation. (a) Average area under the ROC curve, (b) average precision and (c) log diagnostic odds ratio.

For $k = 4$ cross-validation, the computational time for training/adapting per tree τ_{tar} for DA (with three levels of refinement coarse, medium and fine) was computed as 1.671 s. No significant time difference was observed for different variants of DA (± 0.014 s). For the baselines, BL1, BL2, and BL3, the τ_{tar} was computed as 0.563 s, 0.565 s and 0.967 s respectively.

4.5.6. Comparison to state-of-the art domain adaptation methods

In this section, we compare the proposed domain adaptation of random forest framework to other popular domain adaptation methods. These include Frustratingly Easy Domain Adaption (FEDA) proposed by Daumé (2007), Geodesic Flow Kernel (GFK) proposed by Gong et al. (2012), Subspace Alignment (SA) by Fernando et al. (2013) and Joint Cross-Domain Classification and Subspace Learning (JCSL) by Fernando et al. (2015). FEDA is a feature augmentation based domain adaptation method and can be supported by any learning model learnt on the augmented space. We also include additional models based on support vector machine trained on the source domain and tested on the target domain (No Adaptation (NA) – SVM) to act as a baseline for SVM driven state-of-the art methods (SA and JCSL) and nearest neighbor (NN) based classifier (No Adaptation (NA) – NN) to act as baseline for GFK which uses nearest - neighbor classification as a base model. For SVM based methods(NA-SVM, SA and JCSL), the final classifier for TC is a linear SVM with a single hyperparameter weighing classification margin to training error. This hyperparameter is tuned by folded cross-validation ($k = 4$) over the range $\left\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10\right\}$. For NN methods, the number of nearest neighbors to be considered is selected by folded cross-validation ($k = 4$) over a range of 1–10. As the base models of these state-of-the art methods are not ensemble based, for fair comparison we choose a single decision tree base model for FEDA, our baselines and the variants of DA. The mean logDOR and 95% confidence intervals evaluated through $k = 4$ cross-validation over the target domain are illustrated in Fig. 14.

Observations: From Fig. 14, we observe that the variants of DA significantly outperformed the baselines and other domain adaptation methods considered for comparison. Since the task at hand is highly non-linear and class-wise distributions cannot be linearly separated the feature space, we observe that non-linear domain adaptation methods (proposed method, FEDA and GFK) outperform linear adaptation methods (SA and JCSL). Comparing tree-based methods, we observe that FEDA is superior to baselines BL1 and BL2 and has comparable performance with BL3. However, we must note that feature augmentation results in higher computational costs both while training and testing as the augmented feature
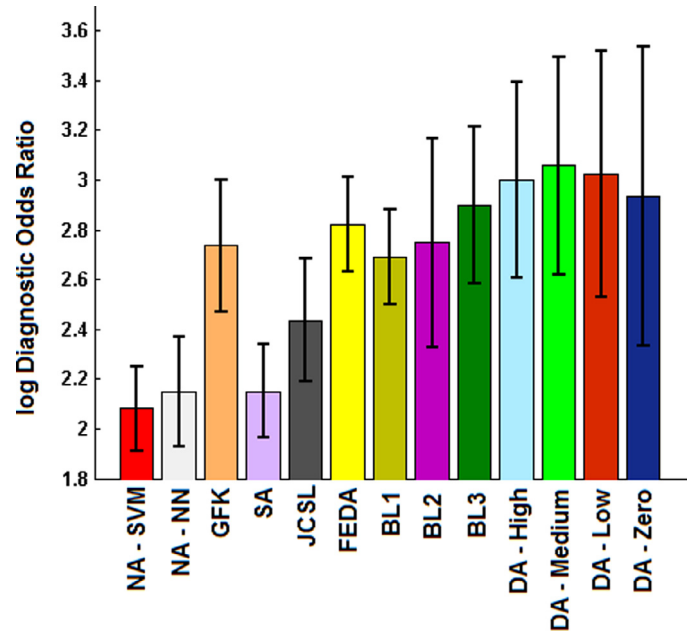
Fig. 14. Comparison to state of the art domain adaptation methods (for $k = 4$ cross-validation and `numTrees` $= 1$). The average testing error with 95% confidence limits evaluated using logDOR metric is plotted for the different comparative methods.

space is of higher dimensionality than the original feature space. All the variants of DA (in particular DA_{Medium}) perform better in comparison to FEDA, implying that the proposed adaptation strategy of model transfer is better than feature augmentation for the task at hand. Additionally, we also observe that for the results for the baselines and variants of DA are consistent with our earlier observations in Section 4.5.2.

5. Conclusions and future directions

In the proposed work, we presented a novel method for supervised domain adaptation of random forests. We work towards the objective of making the forest maximally discriminative on a target domain by leveraging reliably trained models from a closely related source domain and a few labeled exemplars from the target domain. We demonstrated that in presence of a distribution shift, our approach of domain adapting is a superior alternative to methods like performing global domain alignment and directly de-

ploying the source classifier and training a domain-invariant classifier by combining both datasets. In the presence of few examples, the proposed approach is successful in factoring out the effect of domain shift at local and global levels by adapting decision boundaries in a regularized fashion to work on the target domain through the formulation of error-correcting hierarchical transfer relaxation and updating leaf posteriors thorough locally adaptive reweighting.

The exposition of the proposed supervised domain adaptation framework is generic and we present the proof of concept for the challenging scenario of *in vivo* TC of atherosclerotic plaques using intravascular ultrasound. Through the proposed DA method, we successfully adapt models trained on *in vitro* data for *in vivo* deployment. In a broader sense, the contributions in the work in terms of mechanisms for model adaption of random forests can be used towards applications of supervised online learning (leaf node re-weighting) and concept drift (transfer relaxation + leaf node re-weighting), where a formerly trained forest may no longer be valid as the domain evolves over time. We demonstrate that leveraging already available data and models from source domain and adapting with the proposed DA method given a few labeled examples on the target domain is as effective as creating an exhaustive labeled dataset and learning independently for every plausible domain evolution.

To conclude, we effectively establish that domain adapting a reliably trained source forest for the target domain offer a better learning characteristics like higher start, slope and asymptote of the learning curves for the scenario of adapting with few examples. We envisage that this method can be leveraged to reduce annotation and data acquisition costs and improve computational efficiency over conventional retraining approaches.

Acknowledgments

Appendix A. Pseudo-codes

Algorithm 1: Bivariate oblique split (`GenSplit`).

Let:
$\mathcal{D}_n : (X_n, Y_n)$ Training data reaching node n
$\left.\begin{array}{l} \texttt{numSplit} \\ \texttt{numFeat} \end{array}\right\}$ Split parameters
Result: Split function ϕ_n
Initialization: Current candidate index $c = 1$
$\mu_n \leftarrow$ Featurewise mean (X_n);
$\sigma_n \leftarrow$ Featurewise standard deviation (X_n);
repeat
$\quad \alpha_c \leftarrow \text{rand}(\sqrt{\sum |\alpha_c|.^2} = 1)$;
$\quad maxVal \leftarrow \max\ (\alpha_c . * (\frac{\mathbf{x} - \mu_n}{\sigma_n}) | \mathbf{x} \in X_n) minVal \leftarrow$
$\quad \min\ (\alpha_c . * (\frac{\mathbf{x} - \mu_n}{\sigma_n}) | \mathbf{x} \in X_n) \alpha_c^0 \leftarrow minVal + (\text{rand}([0, 1]) *$
$\quad (maxVal - minVal))$ Parameter set: $\theta_c \leftarrow$
$\quad \{\mu_n, \sigma_n, \alpha_c, \alpha_c^0\}$ Information gain: $\mathcal{E}(\theta_c) \leftarrow \mathcal{I}(\theta_c | \mathcal{D}_n)$;
$\quad c \leftarrow c + 1$;
until c reaches `numSplit`;
$c^* \leftarrow \underset{c}{\arg\max}\ \mathcal{E}(\theta_c)$;
$\theta_n \leftarrow \theta_{c^*} \big\}$ Optimal bivariate split;

Algorithm 2: Transfer relaxation (`AdaptS`).

Let:
$\mathcal{D}_{\text{tar}}^n : (X_{\text{tar}}^n, Y_{\text{tar}}^n)$ Target data reaching node n
$\quad \theta_{\text{anchor}}^n \qquad$ Anchor split [arameters
$\quad \texttt{numSplit} \qquad$ Number of candidate splits
$\quad \Gamma \qquad$ Transfer relaxation parameter
Result: Adapted split parameter Set θ_{tar}^n
Initialization: Current candidate index $c = 1$
$\mu_{\text{tar}}^n \leftarrow$ Feature wise mean (X_{tar}^n);
$\sigma_{\text{tar}}^n \leftarrow$ Sign-corrected feature wise standard deviation (X_{tar}^n);
repeat
$\quad$ **Transfer relaxation**
$\quad\ \left.\begin{array}{l} \text{Coefficient: } \alpha_c = \alpha_{\text{anchor}}^n + \mathcal{N}(0, \Gamma) \\ \text{Intercept: } \alpha_c^0 = \alpha_{\text{anchor}}^{n0} + \mathcal{N}(0, \Gamma) \end{array}\right\} \theta_c$ – candidate split
$\quad$ Parameter Set: $\theta_c \leftarrow \{\mu_{\text{tar}}^n, \sigma_{\text{tar}}^n, \alpha_c, \alpha_c^0\}$;
$\quad \mathcal{E}(\theta_c) \leftarrow \mathcal{I}(\theta_c | \mathcal{D}_{\text{tar}}^n) - \mathcal{I}(\theta_{\text{anchor}}^n | \mathcal{D}_{\text{tar}}^n)$;
$\quad c \leftarrow c + 1$;
until c reaches `numSplit`;
$c^* \leftarrow \underset{c}{\arg\max}\ \mathcal{E}(\theta_c)$;
$\theta_{\text{tar}}^n \leftarrow \theta_{c^*} \big\}$ Optimal adapted split;

References

Becker, C., Christoudias, C., Fua, P., 2015. Domain adaptation for microscopy imaging. IEEE Trans. Med. Imaging 34(5), 1125–1139.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J.W., 2010. A theory of learning from different domains. Mach. Learn. 79 (1–2), 151–175.

Ben-David, S., Blitzer, J., Crammer, K., Pereira, F., 2007. Analysis of representations for domain adaptation. In: Proceedings of Advances in Neural Information Processing Systems, vol. 19, pp. 137–144.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Caruana, R., 1997. Multitask learning. Mach. Learn. 28 (1), 41–75.

Chapelle, O., Schlköpf, B., Zien, A., 2010. Semi-Supervised Learning. MIT Press, Cambridge MA, USA.

Cheng, B., Zhang, D., Shen, D., 2012. Domain transfer learning for MCI conversion prediction. In: Proceedings of International Conference on Medical Image Computing and Computer Assisted Intervention. In: Volume 7510 of LNCS, pp. 82–90.

Chopra, S., Balakrishnan, S., Gopalan, R., 2013. DLID: deep learning for domain adaptation by interpolating between domains.. In: Proceedings of ICML Workshop on Challenges in Representation Learning, vol. 2, pp. 5–13.

Ciompi, F., Pujol, O., Gatta, C., Rodríguez-Leor, O., Mauri-Ferré, J., Radeva, P., 2010. Fusing in-vitro and in-vivo intravascular ultrasound data for plaque characterization. Int. J. Cardiovasc. Imaging 26 (7), 762–779.

Criminisi, A., Shotton, J., Konukoglu, E., 2012. Decision forests: a unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. Found. Trends Comput. Graph. Vis. 7, 81–227.

Culjat, M.O., Goldenberg, D., Tewari, P., Singh, R.S., 2010. A review of tissue substitutes for ultrasound imaging. Ultrasound Med. Biol. 36 (6), 861–873.

Daumé, H.I., 2007. Frustratingly easy domain adaptation. In: Proceedings of Annual Meeting of the Association for Computational Linguistics, p. 256.

Escalera, S., Pujol, O., Mauri, J., Radeva, P., 2009. Intravascular ultrasound tissue characterization with sub-class error-correcting output codes. J. Signal Process. Syst. 55 (1–3), 35–47.

Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T., 2013. Unsupervised visual domain adaptation using subspace alignment. In: Proceedings of IEEE International Conference on Computer Vision, pp. 2960–2967.

Fernando, B., Tommasi, T., Tuytelaars, T., 2015. Joint cross-domain classification and subspace learning for unsupervised adaptation. Pattern Recognit. Lett. 65, 60–66.

Ganin, Y., Lempitsky, V., 2015. Unsupervised domain adaptation by backpropagation. In: Proceedings of International Conference on Machine Learning, pp. 1180–1189.

Glas, A.S., Lijmer, J.G., Prins, M.H., Bonsel, G.J., Bossuyt, P.M., 2003. The diagnostic odds ratio: a single indicator of test performance. J. Clin. Epidemiol. 56 (11), 1129–1135.

Goetz, M., Weber, C., Stieltjes, B., Maier-Hein, K., Maier, K., 2012. Learning from small amounts of labeled data in a brain tumor classification task. In: Proceedings of Advances in Neural Information Processing Systems, pp. 1–8.

Gong, B., Shi, Y., Sha, F., Sha, F., 2012. Geodesic flow kernel for unsupervised domain adaptation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 2066–2073.

Goussies, N.A., Ubalde, S., Mejail, M., 2014. Transfer learning decision forests for gesture recognition. J. Mach. Learn. Res. 15, 3667–3690.

Hamm, J., Lee, D.D., 2008. Grassmann discriminant analysis: a unifying view on subspace-based learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 376–383.

Hauberg, S., Feragen, A., Black, M.J., 2014. Grassmann averages for scalable robust PCA. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 3810–3817.

Heimann, T., Mountney, P., John, M., Ionasec, R., 2015. Real-time ultrasound transducer localization in fluoroscopy images by transfer learning from synthetic training data. Med. Image Anal. 18 (8), 1320–1328.

Karamalis, A., Katouzian, A., Carlier, S.G., Navab, N., 2012. Confidence estimation in IVUS radio-frequency data with random walks. In: Proceedings of International Symposium on Biomedical Imaging, pp. 1068–1071.

Katouzian, A., Karamalis, A., Sheet, D., Konofagou, E., Baseri, B., Carlier, S.G., Konig, A., Navab, N., Laine, A.F., 2012a. Iterative self-organizing atherosclerotic tissue labeling in intravascular ultrasound images and comparison with virtual histology. IEEE Trans. Biomed. Eng. 59 (11), 3039–3049.

Katouzian, A., Angelini, E.D., Baseri, B., Carlier, S.G., Suri, J.S., Navab, N., Laine, A.F., 2012b. A state-of-the-art review on segmentation algorithms in intravascular ultrasound (IVUS) images. IEEE Trans. Inf. Technol. Biomed. 16 (5), 823–834.

Katouzian, A., Baseri, B., Konofagou, E.E., Laine, A.F., 2008. Challenges in atherosclerotic plaque characterization with intravascular ultrasound (IVUS): From data collection to classification. IEEE Trans. Inf. Tech. Biomed. 12, 315–327.

Katouzian, A., Laine, A.F., 2010. Methods in atherosclerotic plaque characterization using intravascular ultrasound (IVUS) images and backscattered signals. Atherosclerosis Disease Management, pp. 121–152. doi:10.1007/978-1-4419-7222-4_6.

Katouzian, A., Sathyanarayana, S., Li, W., Thomas, T., Carlier, S.G., 2007. Challenges in tissue characterization from backscattered intravascular ultrasound signals. Proc. SPIE 6513. 65130O-65130O-11.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems, pp. 1097–1105.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444.

Leistner, C., Saffari, A., Santner, J., Bischof, H., 2009. Semi-supervised random forests. In: Proceedings of IEEE International Conference on Computer Vision, pp. 506–513.

Menze, B.H., Kelm, B.M., Splitthoff, D.N., Koethe, U., Hamprecht, F.A., 2011. On oblique random forests. In: Proceedings of Machine Learning and Knowledge Discovery in Databases, pp. 453–469.

Mintz, G.S., 2014. Clinical utility of intravascular imaging and physiology in coronary artery disease. J. Am. Coll. Cardiol. 65 (2), 207–222.

Nair, A., Kuban, B.D., Tuzcu, E.M., Schoenhagen, P., Nissen, S.E., Vince, D.G., 2002. Coronary plaque classification with intravascular ultrasound radiofrequency data analysis. Circulation 106 (17), 2200–2206.

Ohota, M., Kawasaki, M., Ismail, T.F., Hattori, K., Serruys, P.W., Ozaki, Y., 2012. A histological and clinical comparison of new and conventional integrated backscatter intravascular ultrasound (IB-IVUS). Circulation 76 (7), 1678–1686.

Pan, S.J., Kwok, J.T., Yang, Q., 2008. Transfer learning via dimensionality reduction. Assoc. Adv. Artif. Intell. 8, 677–682.

Patel, V.M., Gopalan, R., Li, R., Chellappa, R., 2015. Visual domain adaptation: a survey of recent advances. IEEE Signal Process. Mag. 32 (3), 53–69.

Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y., 2007. Self-taught learning: transfer learning from unlabeled data. In: Proceedings of International Conference on Machine Learning, pp. 759–766.

Rodner, E., Denzler, J., 2009. Learning with few examples by transferring feature relevance. Pattern Recognition. Volume 5748 of LNCS. Springer Berlin Heidelberg, pp. 252–261.

Salperwyck, C., Lemaire, V., 2011. Learning with few examples: an empirical study on leading classifiers. In: Proceedings of International Joint Conference on Neural Networks, pp. 1010–1019.

Sathyanarayana, S., Carlier, S., Li, W., Thomas, L., 2009. Characterisation of atherosclerotic plaque by spectral similarity of radiofrequency intravascular ultrasound signals. EuroIntervention 5 (1), 133–139.

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117.

Schneider, C.A., Rasband, W.S., Eliceiri, K.W., 2012. NIH image to ImageJ: 25 years of image analysis. Nat. Methods 9 (7), 671–675.

Seabra, J.C., Ciompi, F., Pujol, O., Mauri, J., Radeva, P., Sanches, J., 2011. Rayleigh mixture model for plaque characterization in intravascular ultrasound. IEEE Trans. Inf. Technol. Biomed. 58 (5), 1313–1324.

Shankar, P.M., 2001. Ultrasonic tissue characterization using a generalized Nakagami model. IEEE Trans. Ultrason. Ferroelectr. Freq. Control 48 (6), 1716–1720.

Sheet, D., Karamalis, A., Eslami, A., Noël, P., Chatterjee, J., Ray, A.K., Laine, A.F., Carlier, S.G., Navab, N., Katouzian, A., 2014. Joint learning of ultrasonic backscattering statistical physics and signal confidence primal for characterizing atherosclerotic plaques using intravascular ultrasound. Med. Image Anal. 18 (1), 103–117.

Shimodaira, H., 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. J. Stat. Plan. Inference 90 (2), 227–244.

Shotton, J., Sharp, T., Kohli, P., Nowozin, S., Winn, J., Criminisi, A., 2013. Decision jungles: compact and rich models for classification. In: Proceedings of Advances in Neural Information Processing Systems, pp. 234–242.

Sun, S., 2013. A survey of multi-view machine learning. Neural Comput. Appl. 23 (7–8), 2031–2038.

Taki, A., Hetterich, H., Roodaki, A., Setarehdan, S.K., Unal, G., Rieber, J., Navab, N., König, A., 2010. A new approach for improving coronary plaque component analysis based on intravascular ultrasound image. Ultrasound Med. Biol. 38 (8), 1245–1258.

Tanaka, K., Carlier, S.G., Katouzian, A., Mintz, G.S., 2007. Characterization of the intravascular ultrasound radiofrequency signal within regions of acoustic shadowing behind calcium. J. Am. Coll. Cardiol. 49 (9), 29B.

Tommasi, T., Orabona, F., Caputo, B., 2014. Learning categories from few examples with multi model knowledge transfer. IEEE Trans. Pattern Anal. Mach. Intell. 36 (5), 928–941.

Torralba, A., Efros, A.A., 2011. Unbiased look at dataset bias. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1521–1528.

Treeby, B.E., Zhang, E.Z., Thomas, A.S., Cox, B.T., 2011. Measurement of the ultrasound attenuation and dispersion in whole human blood and its components from 0–70 MHz. Ultrasound Med. Biol. 37 (2), 289–300.

van Engelen, A., van Dijk, A.C., Truijman, M.T., van T. Klooster, R., van Opbroek, A., Niessen, W.J., Kooi, M.E., de Bruijne, M., 2015. Multi-center MRI carotid plaque component segmentation using feature normalization and transfer learning. IEEE Trans. Med. Imaging 34 (6), 1294–1305.

van Opbroek, A., Ikram, M.A., Vernooij, M.W., de Bruijne, M., 2015. Transfer learning improves supervised image segmentation across imaging protocols. IEEE Trans. Med. Imaging 34 (5), 1018–1030.

Vezhnevets, A., Buhmann, J.M., 2011. Agnostic domain adaptation. Pattern Recognition. Volume 6853 of LNCS. Springer Berlin Heidelberg, pp. 376–385.

Widmer, G., Kubat, M., 2011. Learning in the presence of concept drift and hidden contexts. Mach. Learn. 23 (1), 69–101.

won Lee, J., Giraud-Carrier, C., 2007. Transfer learning in decision trees. In: Proceedings of International Joint Conference on Neural Networks, pp. 726–731.