Medical Image Analysis xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Encoding atlases by randomized classification forests for efficient multi-atlas label propagation

D. Zikic [a,*], B. Glocker [b], A. Criminisi [a]

[a] Microsoft Research, 21 Station Road, Cambridge CB1 2FB, United Kingdom
[b] Biomedical Image Analysis Group, Imperial College London, 180 Queen's Gate, London SW7 2AZ, United Kingdom

ARTICLE INFO

Article history:
Received 24 January 2014
Received in revised form 23 June 2014
Accepted 25 June 2014
Available online xxxx

Keywords:
Randomized forest
Multi-atlas label propagation
Brain
Segmentation

ABSTRACT

We propose a method for multi-atlas label propagation (MALP) based on encoding the individual atlases by randomized classification forests. Most current approaches perform a non-linear registration between all atlases and the target image, followed by a sophisticated fusion scheme. While these approaches can achieve high accuracy, in general they do so at high computational cost. This might negatively affect the scalability to large databases and experimentation. To tackle this issue, we propose to use a small and deep classification forest to encode each atlas individually in reference to an aligned probabilistic atlas, resulting in an *Atlas Forest* (AF). Our classifier-based encoding differs from current MALP approaches, which represent each point in the atlas either directly as a single image/label value pair, or by a set of corresponding patches. At test time, each AF produces one probabilistic label estimate, and their fusion is done by averaging. Our scheme performs only one registration per target image, achieves good results with a simple fusion scheme, and allows for efficient experimentation. In contrast to standard forest schemes, in which each tree would be trained on all atlases, our approach retains the advantages of the standard MALP framework. The target-specific selection of atlases remains possible, and incorporation of new scans is straightforward without retraining. The evaluation on four different databases shows accuracy within the range of the state of the art at a significantly lower running time.

1. Introduction

Labeling of healthy human brain anatomy is a crucial prerequisite for many clinical and research applications. Due to the involved effort (a fully manual labeling of a single brain takes 2–3 days (Klein and Tourville, 2012)), and increasing database sizes (e.g. ADNI, IXI, OASIS), a lot of research has been devoted to develop automatic methods for this task. While brain labeling is a general segmentation task (with a high number of labels), the standard approach for this task is multi-atlas label propagation (MALP) – see (Landman and Warfield, 2012) for an overview of the state of the art. With the *atlas* denoting a single labeled scan, MALP methods first derive a set of label proposals for the target image, each based on a single atlas, and then combine these proposals into a final estimate.

Currently, there are two main strategies for estimating atlas-specific label proposals. The first and larger group of methods non-linearly aligns each of the atlas images to the target image, and then – assuming one-to-one correspondence at each point – uses the atlas labels directly as label proposals, cf. e.g. (Rohlfing et al., 2004; Warfield et al., 2004; Heckemann et al., 2006). The second group of patch-based methods has recently enjoyed increased attention (Coupé et al., 2011; Rousseau et al., 2011; Wu et al., 2012). Here, the label proposal is estimated for each point in the target image by a local similarity-based search in the atlas. Patch-based approaches relax the one-to-one assumption, and aim at reducing the computational times by using linear instead of deformable alignment (Coupé et al., 2011; Rousseau et al., 2011), resulting in labeling running times of 22–130 min per target on the IBSR dataset (Rousseau et al., 2011). The fusion step, which combines the atlas-specific label proposals into a final estimate, aims to correct for inaccurate registration or labelings. While label fusion is a very active research topic, it is not the focus of this work. Additionally, some approaches perform further refinement, e.g. by learning classifiers for fine-scale class-based correction (Wang et al., 2012).

While current state of art techniques can achieve high levels of accuracy, in general they are computationally demanding. This is primarily due to the *non-linear registration between all atlases and the target image*, combined with the long running times for the best performing registration schemes for the problem (Klein et al.,

[*] Corresponding author.
E-mail address: darko.zikic@tum.de (D. Zikic).

http://dx.doi.org/10.1016/j.media.2014.06.010
1361-8415/© 2014 Elsevier B.V. All rights reserved.

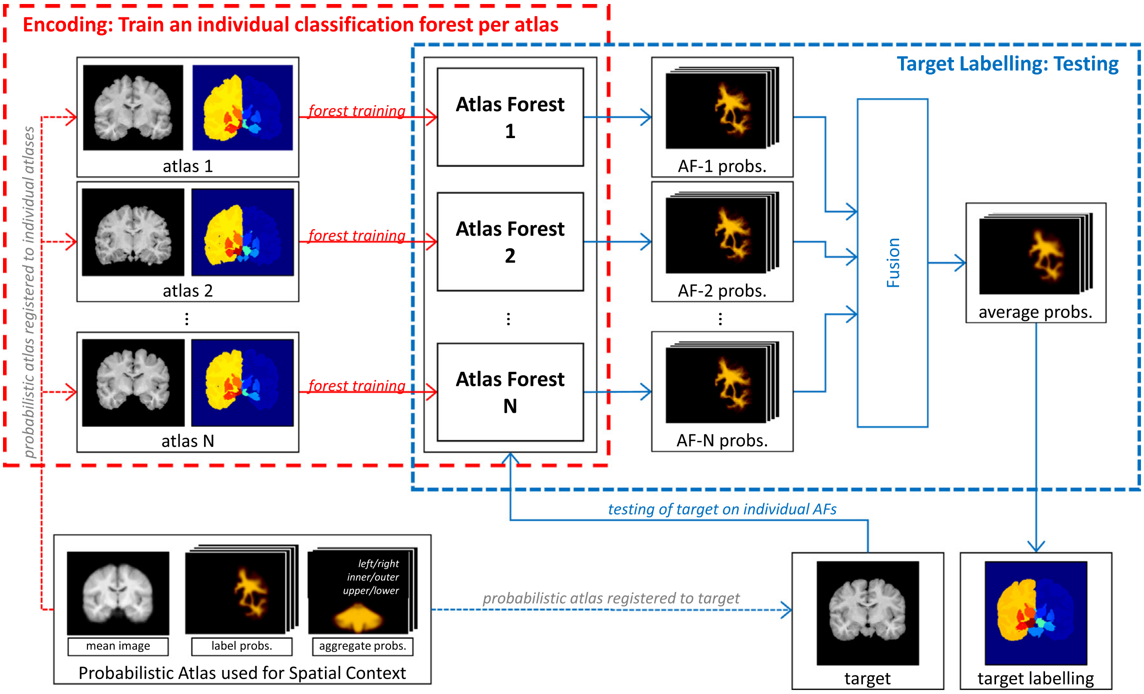

Encoding: Train an individual classification forest per atlas
Target Labelling: Testing
probabilistic atlas registered to individual atlases
atlas 1
atlas 2
atlas N
forest training
forest training
forest training
Atlas Forest 1
Atlas Forest 2
Atlas Forest N
AF-1 probs.
AF-2 probs.
AF-N probs.
Fusion
average probs.
testing of target on individual AFs
left/right
inner/outer
upper/lower
mean image
label probs.
aggregate probs.
Probabilistic Atlas used for Spatial Context
probabilistic atlas registered to target
target
target labelling

Fig. 1. Framework overview. A single atlas is encoded by training a corresponding atlas forest on the samples from that atlas only. The labeling of a new target is performed by the testing step on the trained atlas forests, and the following fusion of the probabilistic estimates by averaging. For the entire method, the intensity images are augmented by label priors as further channels, obtained by registering a probabilistic atlas.

2009). Current methods state running times of 2–20 h per single registration (Landman and Warfield, 2012). Furthermore, sophisticated fusion schemes can also be computationally expensive. State of the art approaches report fusion running times of 3–5 h (Wang et al., 2012; Asman and Landman, 2012a; Asman and Landman, 2012b).

While the major drawback of high computational costs is the scalability to large and growing databases, they also limit the amount of possible experimentation during the algorithm development phase.

Our method differs from previous MALP approaches in the way how label proposals for a single atlas are generated, and is designed with the goal of low computational cost at test time and experimentation. In this work, we focus on the question of how a single atlas is encoded. From this point of view, methods assuming one-to-one correspondence represent an atlas directly as an image/label-map pair, while patch-based methods encode it by a set of localized patch collections. Variations of the patch-based encoding include use of sparsity (Wu et al., 2012), or use of label-specific *k*NN search structures (Wang et al., 2013).

In contrast to previous representations, we encode a single atlas together with its relation to label priors by a small and deep classification forest – which we call an *Atlas Forest* (AF). Given a target image as input (and an aligned probabilistic atlas), each AF returns a probabilistic label estimate for the target. Label fusion is then performed by averaging the probability estimates obtained from different AFs. Please see Fig. 1 for an overview of our method. While patch-based methods use a static representation for each image point (i.e. a patch of fixed size), our encoding is spatially varying. In the training step, our approach learns to describe different image points by differently shaped features, depending on the point's contextual appearance.

Compared to current MALP methods, our approach has the following important characteristics:

1. *Only one registration per target is required.* This registration aligns the probabilistic atlas to the target. Since only one registration per target is required, the running time is independent of the database size in this respect. This differs conceptually from patch-based approaches, where the efficiency does not come from reducing the number of registrations, but from using affine instead of non-linear transformations.
2. *Efficient generation of atlas proposals and their fusion.* For proposal generation one AF per atlas is evaluated. Due to the inherent efficiency of tree-based classifiers at test time, this is significantly more efficient than current approaches.
3. *Efficient Experimentation.* A leave-one-out cross-validation of a standard MALP approach on n atlases requires registration between all images, thus scaling with n^2. In contrast, the training of the single AFs, which is the most costly component of our approach for experimentation, scales with n (this assumes that generating the probabilistic atlas is not part of experimentation).

Besides being efficient, experiments on 4 databases in Section 3 indicate that our scheme also achieves accuracy within the range of the state of the art.

Being based on discriminative classifiers, our approach is also related to a number of works which employ machine learning techniques. Compared to the use of multi-atlas label propagation techniques discussed above, the use of machine learning for brain labeling is still relatively limited. In (Tu et al., 2008), a hybrid model is proposed, which combines a discriminative probabilistic-boosting tree (PBT) classifier (Tu, 2005) with a PCA-based generative shape model of the individual anatomical structures. In (Tu and Bai, 2010), the Auto-Context framework with the PBT classifier was applied to brain labeling, and shown to outperform (Tu et al., 2008). Recently, the use of classifiers to correct systematic mistakes of labeling methods in a post-processing step has been shown to improve accuracy (Wang et al., 2011, 2012).

The major difference of these works to our approach is that they use the common scheme in which all available atlases are used for the training of one classifier. This is also true of standard forest schemes (cf. e.g. (Shotton et al., 2011; Iglesias et al., 2011a; Montillo et al., 2011; Zikic et al., 2012)) which train each tree on data from *all* training images.

In contrast, *the main idea of this paper is to use one classifier to encode a single atlas by training it only on this exemplar*. This approach has three advantageous properties for the multi-atlas label propagation setting.

1. *Simple incorporation of new atlases into the database.* For standard forest schemes, addition of new training data requires complete retraining or approximations. In our scenario, a new forest is simply trained on the new atlas exemplar and added to the other, previously trained AFs.
2. *Selection of atlases for target-specific evaluation is straightforward* since every AF is associated with a single atlas. This property allows use of atlas-selection (Aljabar et al., 2009), which can improve accuracy and reduce the computational cost. This step seems non-obvious for standard forest schemes where predictions are not separable with respect to specific atlases.
3. *Efficient experimentation.* For cross-validation, standard schemes have to be trained for every training/testing split of data, which is extremely costly. In our scenario, each AF is trained only once. Any leave-k-out test is performed simply by using the subset of $n - k$ AFs corresponding to the training data. This point can be seen as a generalization of the corresponding experimentation efficiency property in the MALP setting.

In general, training ensemble classifiers on *disjunct* subsets of data cannot be expected to reach higher accuracy than training each classifier on all data or overlapping subsets, especially if the subsets are different atlases. The difference in accuracy between the two models will depend on the application, and especially the similarity of the atlases to each other. Furthermore, in practice, the computational complexity of each model will also limit the possibility to set the parameters of each model, such that it performs as close as possible to its theoretical limit. In Section 3.1.2, we experimentally show that the accuracy of the proposed scheme and a ’reasonable’ standard forest scheme seems to be on approximately the same level for the brain labeling task.

The main idea of thinking about a single atlas as a classifier is already mentioned for example in (Rohlfing et al., 2005). And indeed, the action of a single warped atlas in a standard MALP setting is that of a classifier – however a very simple one: For each spatial point the warped atlas will assign the value from the corresponding warped atlas label map.

In this work, we propose the use of non-trivial machine learning-based classifiers to encode individual atlases in the MALP setting, and demonstrate that this approach exceeds the standard encoding in terms of efficiency, while maintaining high accuracy, but also has the additional advantages in comparison to standard learning schemes, as discussed in detail above.

Our work on atlas forests was originally presented in a form of a conference paper in (Zikic et al., 2013a). This article extends the previous conference publication by providing a new evaluation with a simplified system, and a detailed evaluation and analysis of the method, as well as a hopefully improved overall presentation. To our best knowledge, the only other work which considers the use of non-trivial classifiers which are trained by individual atlases is (Akhondi-Asl and Warfield, 2013). The focus of that work is on a generalization of the STAPLE fusion method (Warfield et al., 2004) to operate on probabilistic estimates rather than thresholded label estimates. To generate per-atlas probabilistic estimates, (Akhondi-Asl and Warfield, 2013) uses a Gaussian Mixture Model (GMM) of patch intensities, and trains an individual GMM for each atlas. This article has a focus on efficiency and the relation of the proposed scheme to existing machine learning schemes. It differs from previous work in technical details through use of a different classifier in combination with probabilistic atlases, and a simple averaging of probabilities as the fusion method. After describing the details of the method in the next section, we evaluate its performance and analyze it in Section 3, and discuss and summarize its properties in Section 4.

2. Method – Atlas Forests

An atlas forest (AF) encodes a single atlas by training one randomized classification forest (Breiman, 2001) exclusively on the data from the atlas. Every point in the atlas is described by its (contextual) appearance only, without considering its location (this can be seen as an even further relaxation of the one-to-one assumption, compared to patch-based approaches).

While this allows us to avoid registration of atlases to the target image, a problem with such a location-oblivious approach is that the location of points carries valuable information about label probabilities (e.g. a point on the far left is unlikely to carry a right-side label), see Fig. 2. To efficiently integrate spatial awareness, we augment the intensity information of the images by label prior maps P_L obtained from a registered probabilistic atlas. The prior maps are then treated as additional image channels. The atlas forest then operates during training and testing on this augmented input data. For the alignment of the priors, only a *single* registration per image is required.

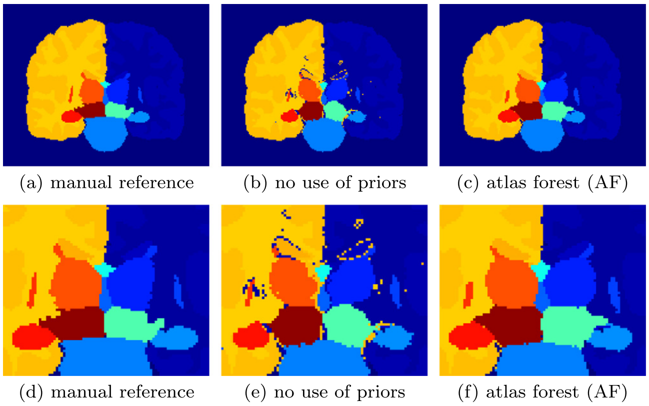
We use randomized forests as a classifier since they can efficiently handle a high number of classes, which is important in the MALP setting. However, any other appropriate classifier might be equally well used. In this paper, we give only the specifics of the used randomized forests – for more details and background, see for example (Criminisi and Shotton, 2013). Classification forests consist of a set of trees, and as a learning-based method, they operate in two stages: training and testing.

2.1. Tree training

During training, each binary tree t in the atlas forest A_i is trained on the specific i-th atlas, which consists of an intensity image I_i and the corresponding label map L_i which contains class labels c. The intensity image is further augmented by label priors as further channels to form a new multi-channel image $\widetilde{I}_i$ (see Section 2.3). Specifically, each tree t learns a class predictor $p_t(c|f)$ for a high-dimensional feature representation f of points from $\widetilde{I}_i$.

The training involves separating (or splitting) the training examples at each node based on the features and with respect to an objective function. The split functions are determined by maximizing the *information gain* at each node for a subspace of the whole feature space.

The feature subspace at each node consists of a set of deterministic features which are considered at every node (local readout in the intensity and label prior channels), and a number of random features, which are instantiated by randomly drawing parameters for the employed feature types – please see Section 2.3 for details. In principle, a certain number (n_f) of different random features are chosen at each node, such that the actual overall dimensionality of the feature space considered during the training of one tree is approximately n_f multiplied with the number of inner nodes in the trained tree. In our actual implementation, the following modification is made. For the first 10 levels, for each level we randomly draw 10 batches with n_f features each. Then, each node at this level randomly selects one of the batches and operates on those



(a) manual reference (b) no use of priors (c) atlas forest (AF)

(d) manual reference (e) no use of priors (f) atlas forest (AF)

Fig. 2. Labeling example (IBSR): Using only intensity-based features leads to extreme errors (b), which can be removed by additional use of label priors (c). Corresponding close-ups are shown in (d,e,f).

features. This reduces running time while not negatively affecting the accuracy. For the experiments, we use $n_f = 500$.

Please note that each tree has access to a different feature subspace. To keep the number of samples as high as possible for training, we use all atlas samples for each tree, i.e. we do not use a bagging strategy.

At each node, we use split functions which consider one-dimensional features (also denoted as axis-aligned), and the optimization is performed by a grid search, independently along each dimension. For each dimension of the feature subspace considered at a given node, we determine the range of values along that dimension for the samples within the node, and uniformly distribute a certain number of thresholds along the estimated range ($n_{\text{thresholds}} = 20$). Then, for evaluated features and all corresponding thresholds, we perform putative splits of the samples into left and right child, and select the combination of feature and threshold which leads to the largest information gain.

Since we are dealing with a high number of unbalanced classes with varying sample sizes, we use class re-weighting for training, i.e. we adjust the probability computation for each class according to its global frequency, such as to obtain a uniform distribution at the root node. Without this step, small classes would have low influence on the split functions, resulting in reduced accuracy for these classes.

Training is stopped at a certain tree depth ($d = 40$), and by the condition that a tree leaf must not contain less than a certain number of samples ($s_{\min} = 8$).

After training, each leaf l contains a class predictor $p^l(c|f)$, which is computed as the re-weighted empirical class distribution of its incoming training samples.

2.2. Labeling by tree testing and fusion

At testing, a target image I is labeled by aligning the probabilistic atlas to it, and then processing the points of the augmented input $\tilde{I}$ using the trained AFs. By applying the learned splitting functions to the feature representation f of a point to be labeled, each tree t from a certain AF yields a prediction $p_t(c|f)$.

The probabilistic estimate of the AF a with n_t trees is then formed as the average of all tree predictions

$$p_a(c|f) = \frac{1}{n_t}\sum_{i=1}^{n_t} p_{t_i}(c|f). \qquad (1)$$

The fusion of these probabilistic estimates from n_a AFs is done by averaging, i.e.

$$p(c|f) = \frac{1}{n_a}\sum_{i=1}^{n_a} p_{A_i}(c|f), \qquad (2)$$

and subsequent maximum selection $\hat{c} = \arg\max_c p(c|f)$.

2.3. Features

To describe an image point at a certain location x, we use at each node a set of deterministic local features and randomly instantiated non-local features, which are selected at each node by supplying specific feature-type functions with randomly drawn parameters.

The deterministic features are local intensity readouts $\tilde{I}(x)$ in a multi-channel image $\tilde{I}$, which is formed by augmenting the atlas image I by the aligned label priors P_L. We refer to this feature set as deterministic, because it is accessible to every node of every tree during training. Next to the priors for the individual labels, we employ further 6 aggregate priors, which contain priors for left/right, lower/upper and inner/outer labels, thus subdividing the brain in a coarser manner. In a setting with $|L|$ different labels, this results in a $|L| + 7$-channel image $\tilde{I}$. The use of the prior labels allows us to include the available knowledge about the label probabilities at this point in an efficient way, at the cost of a *single registration per target*. For an effect of using the label priors, please see Fig. 2. For the statistics of the use of the label priors during the training procedure, please see Fig. 12.

The randomized features at each node are generated by randomly drawing parameters for the feature-type functions. We use the randomized features only on the intensity images, since the combination of the large number of classes and the high-dimensional feature space spanned by the feature-types would not be computationally practical. We describe the intensity around a certain location by a set of intensity-based parametric feature-types, which are non-local but short-range. Given the point of interest x in spatial domain of image I, offset vector $u \in \mathbb{R}^3$, cuboids

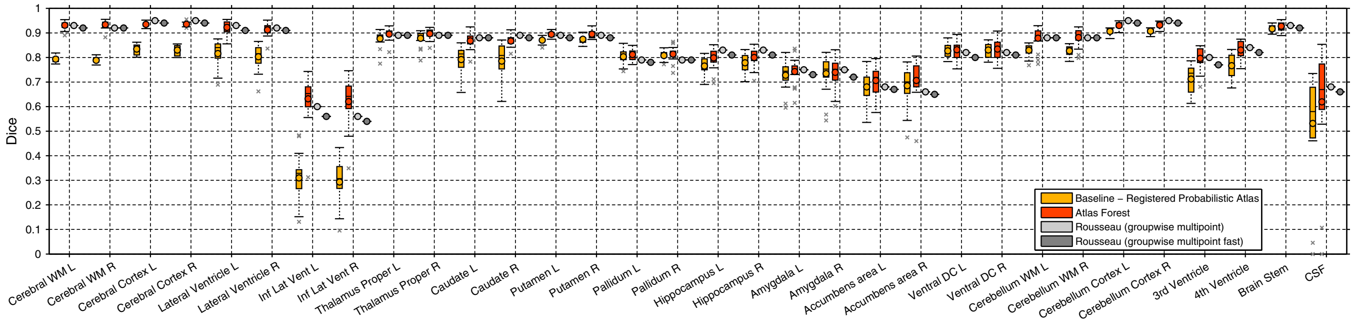


Fig. 3. Leave-1-out cross-validation results on the IBSR database. The summary of the results is given in Table 1 as **AF (non-lin reg)**.

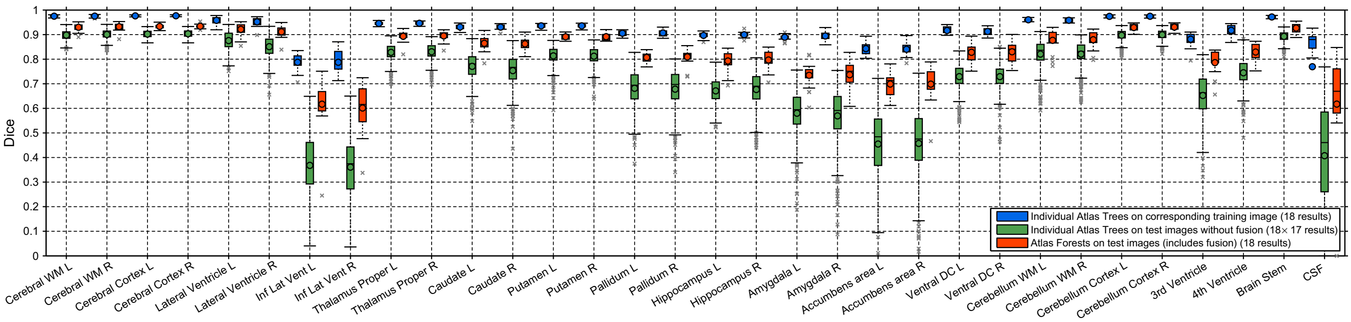


Fig. 4. Evaluation of accuracy of individual trees on testing data (green), and comparison to the actual AF results, i.e. results after fusion by averaging the individually estimated class probabilities (red). Additionally, we evaluate the accuracy of individual trees on corresponding training data (blue). The discrepancy in performance between training and testing (blue vs. green) indicates the amount of overtraining. Note that in our experiments, the analyzed modifications of the system which lead to reduction of training error also reduce the testing error, ultimately leading to worse accuracy. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$C_s(x)$ (centered at x with side lengths $s \in \mathbb{R}^3$), and the mean operator μ, we use the following feature types:

1. Local cuboid mean intensity:

$$F^1_u(I,x) = \mu(I(C_s(x))) \tag{3}$$

2. Difference of local intensity and offset cuboid mean:

$$F^2_{u,s}(I,x) = I(x) - \mu(I(C_s(x+u))) \tag{4}$$

The feature type and its parameters (u,s) are drawn during training at each node uniformly from a predefined range, thus defining the random feature space dimensions to be explored. Guided by the results from patch-based works (Coupé et al., 2011; Rousseau et al., 2011), we use a maximum offset of 15mm, and cuboid side length $s_k < 5$ mm.

Table 1
Average mean and standard deviation of Dice score for the variations discussed in Section 3.1. The results of the proposed method with the chosen settings are repeated with a highlighted name for easier comparison. A visual representation of the results is given in Fig. 5.

Method	Dice mean	Dice σ
Rousseau (GW-MP)	83.5	–
Rousseau (GW-MP fast)	82.3	–
AF (no prob. atl.)	71.6	9.6
AF (no prob. atl.) S-2	78.6	5.7
AF (affine reg.)	80.3	5.9
AF (affine reg.) S-2	80.5	5.5
AF (non-lin reg)	83.5	4.2
AF (non-lin reg) S-2	83.0	4.2
AF (det. features only)	80.2	4.6
AF (GT masks)	84.4	4.2
Standard Forest (6% subs.)	81.7	3.9
Standard Forest (12% subs.)	83.3	3.8
Standard Forest (grid subs.)	82.5	3.8
AF MS-02 T-5	83.3	4.7
AF MS-04 T-5	83.7	4.3
AF MS-08 T-5	83.5	4.2
AF MS-16 T-5	82.5	4.1
AF MS-32 T-5	80.9	4.3
AF MS-08 T-1	83.1	4.1
AF MS-08 T-2	83.4	4.1
AF MS-08 T-3	83.4	4.1
AF MS-08 T-4	83.5	4.1
AF MS-08 T-5	83.5	4.2
Prob. Atlas (Affine-Reg)	65.8	7.2
Prob. Atlas (NL-Reg)	76.8	4.5

2.4. Generation of the probabilistic atlas

We use a probabilistic atlas which consists of an average intensity image $\bar{I}$ and a set of $|L|$ label priors P_L. In this work, we construct simple label priors ourselves since we deal with varying labeling protocols – for actual applications, a use of carefully constructed, protocol-specific priors would seem beneficial, e.g. (Shattuck et al., 2007; Rohlfing et al., 2010). The construction is performed by iterative registration of the training images to their mean (Joshi et al., 2004). This results in an average intensity image $\bar{I}$, and a set of label priors P_L which are created by applying the computed warps to corresponding label maps followed by averaging. We use affine registration, followed by a deformable registration by the FFD-based method from (Glocker et al., 2008).[1] with cross-correlation as data term, and conservative deformable settings with an FFD-grid spacing of 30 mm on the finest level and strong regularization. The registration uses an image pyramid with downsampling factors of 8–2, and takes approximately 3 min per image.

At test time, the average intensity image $\bar{I}$ is registered to the target, and the computed transformation is used to align the label priors P_L to the target. Here, the same registration scheme as above is employed.

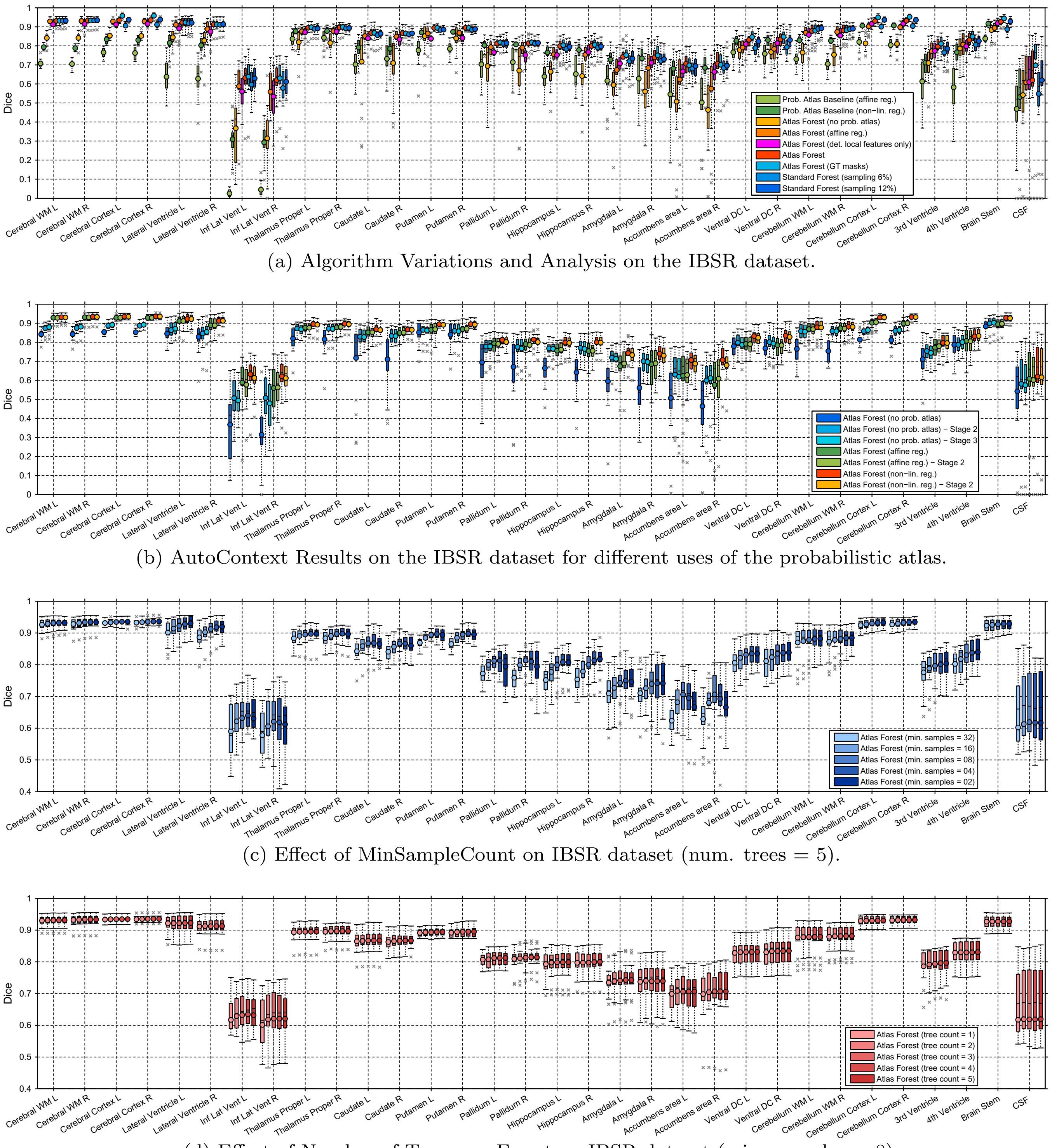
2.5. Auto-context variation

As a variation of the proposed system, we consider using atlas forests within the auto-context meta-framework of (Tu and Bai, 2010). This means running multiple stages of atlas forests, such that the probabilistic output of one stage is used as the label prior for the next one. We initiate the process by using the priors from the probabilistic atlas in the 1st stage, in the same way as for the basic atlas forest method. While the original motivation for auto-context is the regularization of results, in this work we use it to evaluate the possibility of removing the dependency on the registration scheme.

One practical issue with auto-context is the correct use of training data for the different stages. If the same training data is used for all stages, then the probabilistic output of the first stage will have a too high accuracy due to the fact that the testing (which generates the probabilistic output) was performed on an image from the training data set. In consequence, this presents the classifier at the 2nd stage with overconfident probabilities for training, which are not comparable to the ones at test time. Ultimately, this leads to a decreased performance of the system. The correct management of training data within the auto-context scheme is

[1] An implementation is available at http://mrf-registration.net.

Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010



(a) Algorithm Variations and Analysis on the IBSR dataset.

(b) AutoContext Results on the IBSR dataset for different uses of the probabilistic atlas.

(c) Effect of MinSampleCount on IBSR dataset (num. trees = 5).

(d) Effect of Number of Trees per Forest on IBSR dataset (min. samples = 8).

Fig. 5. We analyze the influence of the different method components (a), and the application of an auto-context-type scheme (b), as well as the variation of the minimal allowed sample count per leaf (c), and the number of trees used per atlas forest (d). The quantitative summary of the results is given in Table 1.

much easier to achieve with the AF framework than with the standard forest scheme. It can be simply done by excluding the i-th atlas forest A_i for the generation of the priors for the i-th training image – in the same way as this is done for leave-1-out validation.

3. Evaluation and analysis

We evaluate our approach on four brain MRI data sets:

– IBSR database (Section 3.1).
– LPBA40 database (Section 3.2).
– MICCAI 2012 Multi-Atlas Labeling Challenge (3.3).
– MICCAI 2013 SATA Challenge (Section 3.4).

Additionally, we perform an analysis of the influence of the different method components and their variations in Sections 3.1.1, 3.1.2, 3.1.3 and 3.1.4, and analyse the structure of the trees trained

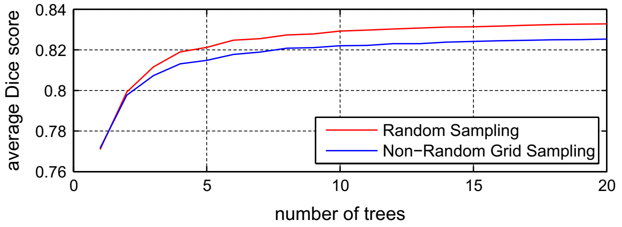


Fig. 6. Accuracy as a function of the number of trees, for the standard forest with randomized subsampling (rate 12%), and deterministic sampling on a grid with a step size of 2 along each dimension (rate 12.5%).

by our method in Section 3.5, both on the data from the IBSR database.

For all tests we perform the standard preprocessing steps in the following order:

- skull-stripping,
- inhomogeneity correction (Tustison and Gee, 2010),
- histogram matching (www.itk.org).

The computation of brain masks for the skull-stripping is done differently for the different data sets. Only points within the mask are used for training and testing. For histogram adaptation, we perform matching to the histogram of the first image in each atlas library as reference.

We used the IBSR dataset for the development of the method and the estimation of the parameters. All subsequent experiments are performed with the same fixed settings. In the final settings, we use 5 trees per atlas forest, and tree growth is stopped primarily by the criterion which restricts the minimal number of samples per leaf to 8. For practical reasons, the tree depth is limited to 40. At training time, each node in a tree considers $n_f = 500$ random features and a set of local readouts on each of the input channels (intensity and label priors from the registered probabilistic atlas) to determine the split functions.

Training was done on several single PCs with different specifications. The average training time for one tree is ca. 10–30 min, depending on the exact hardware and the number of classes in the experiment. For testing, we report the running times observed on a single desktop PC (Intel Xeon E5520 2.27 GHz, 12 GB RAM). Across the experiments, the test running times are in the range of 2–8 min per target image. These times depend linearly on the number of atlases and the number of trees per atlas forest. The running time also depends on the number of class labels for the problem at hand. The reported testing times are for the label propagation only, and do not include the time for the pre-processing of the image, or the registration of the probabilistic atlas and the corresponding warping of the label priors (ca. 3–5 min). The wall-clock time for the labeling of one target image is thus in the range of 5–13 min.

3.1. IBSR database

The IBSR data (http://www.nitrc.org/projects/ibsr) contains 18 labeled T1 MR images. In this work we use the set of 32 primarily subcortical labels. For skull-stripping, we use the brain masks which are provided with the dataset. With the above settings our approach reaches a mean Dice score of 83.5 ± 4.2%, while requiring ca. 2 min for the evaluation of the atlas forests per target image. To provide a comparative context, we cite the results from (Rousseau et al., 2011), which are considered state of the art on this data set. The IBSR data set is used in (Rousseau et al., 2011) in a leave-one-out evaluation, and the best performing version of the proposed method (group-wise multipoint (GW-MP)) reaches a mean Dice of 83.5%, with a running time of 130 min. A different variant discussed in (Rousseau et al., 2011) (group-wise fast multipoint (GW-MP fast)), which aims at faster running times by performing the search at a reduced number of locations in the image, reaches a Dice of 82.3%, with a labeling time of 22 min. The results of this experiment are presented in Fig. 3.

Further, we use the IBSR data to evaluate variations of our method discussed below, all tested by leave-one-out experiments.

3.1.1. Influence of method components

In this section we study the influence of the different components of our method – the results are summarized in Figs. 5a and 4, and Table 1.

There is a clear increase in accuracy from not using a probabilistic atlas (71.6 ± 9.6%), to using an affinely registered probabilistic atlas (80.3 ± 5.9%), to using a non-linearly registered atlas as done in the proposed method (83.5 ± 4.2%). For completeness, we also show the performance of using a probabilistic atlas alone (without running any trained classifier) as a baseline, with affine (65.8 ± 7.2%) and non-linear registration (76.8 ± 4.5%).

Further, we study the contribution of the deterministic and randomized features. To this end, we train one tree per AF, with deterministic features only, which leads to Dice scores of 80.2 ± 4.6%. While the additional use of randomized features provides a clear improvement in accuracy (83.5 ± 4.2%), this experiment indicates that a careful design of deterministic non-local features might result in good accuracy with an even higher efficiency. This experiment also provides insight to why the number of trees does not influence the accuracy strongly in the current implementation, cf. Fig. 5d.

In Fig. 4, we show the effect of fusion on the accuracy, in comparison to the predictions of individual trees on testing data. Also, we compare the accuracy of individual tree predictions for training and testing data. The observed difference in accuracy indicates how well tuned the individual trees are to the corresponding atlases, thus indicating the amount of overtraining.

We also evaluate the effect of the quality of the brain masks. Using “ground truth” masks (GT masks), which are computed from the label map increases the accuracy to (84.4 ± 4.2%), indicating room for improvement.

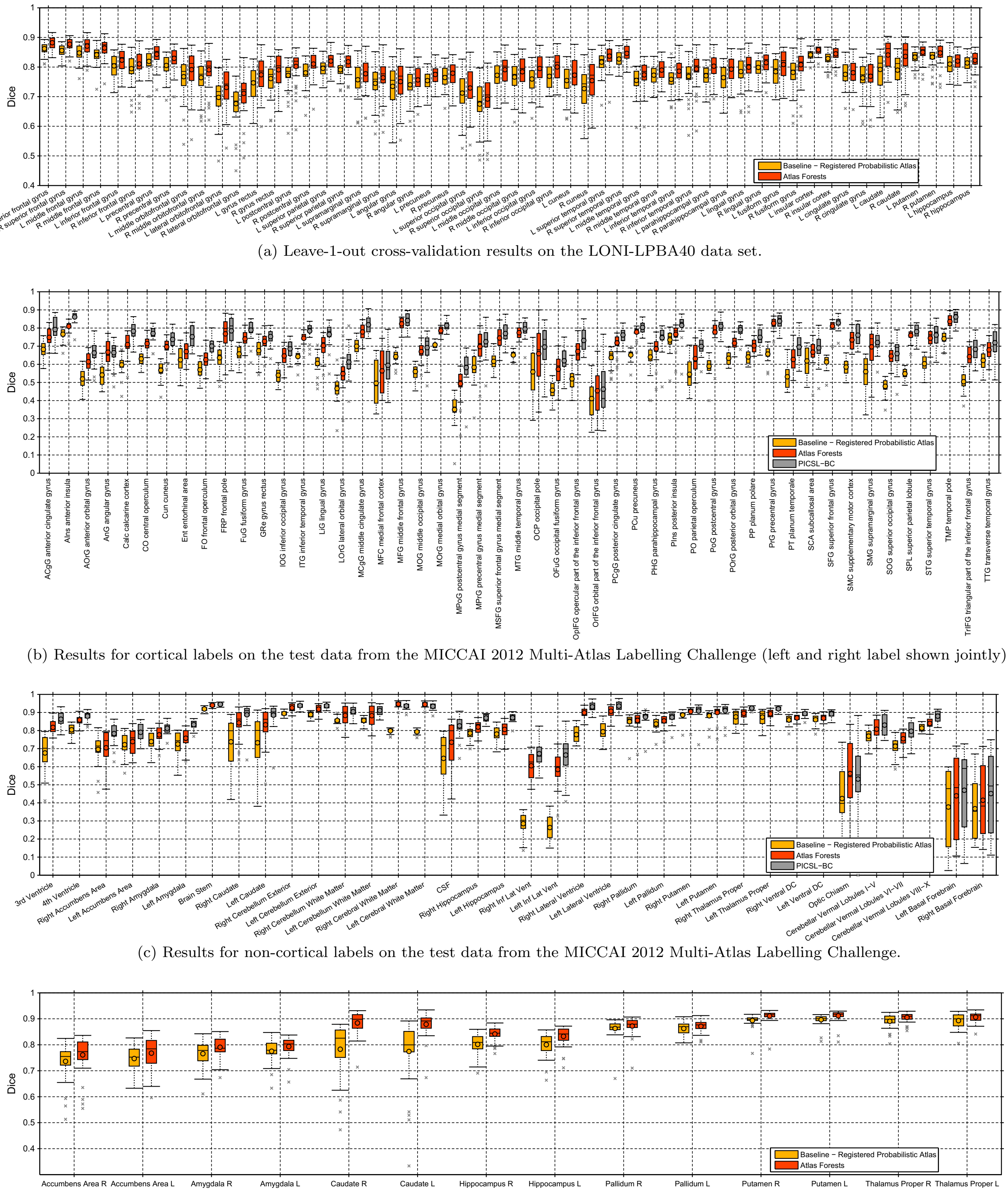
3.1.2. Comparison to the standard forest scheme

Here, we evaluate the performance of a “standard” forest scheme. As previously mentioned, generally, training each classifier of an ensemble on a disjunct subset of data (proposed method) cannot be expected to perform better in terms of accuracy than training each classifier on all data, or overlapping subsets thereof (standard scheme without or with bagging). In practice however, the computational complexity of each model limits the possibility to set its parameters, such that it performs as close as possible to its theoretical limit. Further, the difference in accuracy will depend on the problem at hand.

As it is not possible to devise a perfectly fair comparison between two methods, the following represents our best effort to provide a comparison to a standard forest scheme, which is ’reasonably’ designed within the limits posed by the higher computational requirements of this model. To this end, for the standard forest scheme, we use the same settings as for the AF scheme, with following exceptions.[2] Instead of using all data from all images, we apply a standard bagging strategy in which each tree has access to a

[2] We use the settings determined for the AF scheme, due to the high computational cost of experiments required to tune the parameters of the standard model. The difficulty for experimentation for the standard model is one of the major motivation points for this work, and one of the advantages of the proposed scheme.

Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010



(a) Leave-1-out cross-validation results on the LONI-LPBA40 data set.

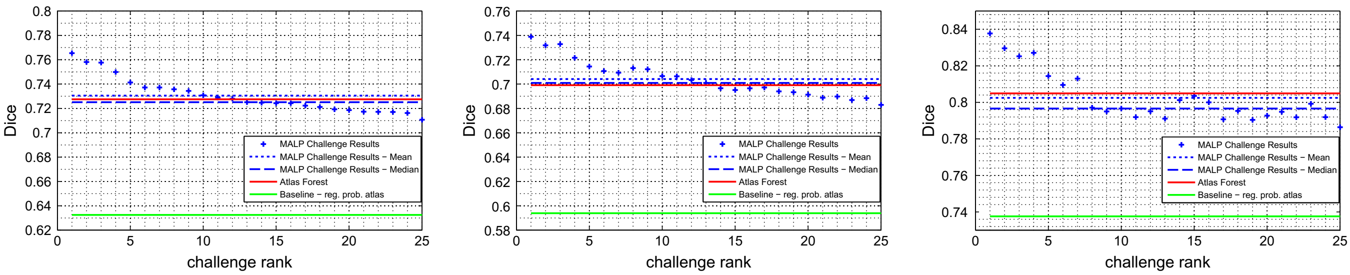
(b) Results for cortical labels on the test data from the MICCAI 2012 Multi-Atlas Labelling Challenge (left and right label shown jointly).

(c) Results for non-cortical labels on the test data from the MICCAI 2012 Multi-Atlas Labelling Challenge.

(d) Leave-1-out cross-validation results on the training data from the MICCAI 2013 SATA Challenge Workshop.

Fig. 7. Summary of the results for different data sets.

subset of the training data. This reduces the high computational burden of the standard scheme to a manageable level, and further has the effect of decorrelating the individual trees. We perform uniform sampling within the brain masks, and perform experiments with two different subsampling rates. First, we use a subsampling rate such that each tree uses approximately the same amount of data for training as in our approach ($\lceil 100\%/(18-1) \rceil = 6\%$). Second, to establish the ability of the standard forest scheme to provide higher accuracy if given more data, we additionally use a subsampling rate of 12%. Finally, to exclude the possibility that the accuracy of the standard forest is negatively influenced by the bagging strategy (which is not used for the atlas forest), we perform an experiment



(a) All structures (b) Cortical structures (c) Non-cortical structures

Fig. 8. Our results in the context of the MICCAI 2012 Multi-Atlas Labeling Challenge results.

in which the samples from each image are chosen from a deterministic regular grid. Here, we use a step size of 2 in each dimension, resulting in a sampling rate of 12.5%.

Each standard forest (one for each leave-1-out experiment) uses 20 trees (this setting is again chosen due to computational budget, and is comparable to the AF setting with 1 tree per forest). The analysis of accuracy depending on the number of trees per forest shows that 20 trees are sufficiently close to the asymptotical state, please see Fig. 6. The results (81.7 ± 3.9% for 6% subsampling rate, and 83.3 ± 3.8% for 12% subsampling rate, and 82.5 ± 3.8% for the deterministic grid sampling) indicate that the data separation in Atlas Forests does not degrade the accuracy compared to the standard forest approach. Please see also Table 1 and Fig. 5a.

3.1.3. Auto-context variation

We test the auto-context variation of the method (Fig. 5b) for the three different usages of the probabilistic atlas. The second auto-context stage is denoted by (S-2). While there is a clear improvement from using the second stage if no probabilistic atlas is used, we do not observe a similar effect when either an affinely or a non-linearly registered probabilistic atlas is used. However, we do observe a slight improvement of the results by applying the auto-context scheme together with the use of a non-linearly registered probabilistic atlas in our original participation in the MICAI 2013 SATA Challenge (where we used slightly different settings of the system) (Zikic et al., 2013b).

3.1.4. Parameter settings

We test the influence of different settings for the minimal allowed number of samples per leaf and subsequently for the number of trees per atlas forest.

For the minimal number of samples per leaf, we find that decreasing this parameter down to 8 or 4 samples improves the accuracy compared to more conservative settings of 32 or 16. Setting this parameter to 2 starts to show indications of overtraining on some classes (e.g. Inf Lat Vent, Accumbens Area), cf. Fig. 5c. For this experiment, we allow trees to grow up to depth 60 to accommodate for the small setting of the minimal sample count parameter. Based on the results of this experiment, we set the minimal sample count to $s_{\min} = 8$ for further experiments.

Next, with fixed $s_{\min} = 8$, we test the influence of the number of trees per atlas forest (Fig. 5d). The performance is stable for different values of this parameter, and we see no large differences between using 1 and 5 trees per atlas forest. This effect is probably due to the use of the deterministic features. We choose to use $T = 5$ as a conservative setting for subsequent experiments.

3.2. LONI-LPBA40 database

The LONI-LPBA40 database (Shattuck et al., 2007) consists of 40 images of healthy volunteers, with 56 labels, most of them within the cortex. After excluding the cerebellum and the brainstem from the set of labels – as these structures are not included in the provided skull-stripped MR images – we end up with 54 labels. Because the MR images are available only in a skull-stripped format, we do not compute the brain masks ourselves for this dataset, but derive them from the image voxels with values larger than 0. Our approach reaches an average Dice of 80.14 ± 4.53%, while the baseline yields 77.91 ± 4.28%. The evaluation of the atlas forests takes ca. 6 min per image. To provide some context, we cite the recent results on this dataset from (Wu et al., 2012), where three methods are evaluated for 54 labels[3]: an implementation of a patch-based scheme as in (Coupé et al., 2011; Rousseau et al., 2011) (PBL), and two modifications aiming at sparsity of used patches (SPBL), and spatial consistency (SCPBL). The corresponding reported Dice scores for a leave-one-out experiment are 75.06%, 76.46% and 78.04%, with running times of 10, 28 and 45 min *per class*.

3.3. MICCAI 2012 Multi-Atlas Labeling Challenge

The data from the MICCAI 2012 Multi-Atlas Labeling Challenge (Landman and Warfield, 2012) consists of 15 training and 20 test T1 MR images from the OASIS project and corresponding label maps as provided by Neuromorphometrics, Inc. (http://Neuromorphometrics.com/) under academic subscription. The dataset has 134 labels (98 cortical, 36 non-cortical). The challenge evaluation system is no longer active and the reference segmentations for the test data set are freely available, as well as the segmentations submitted to the challenge. We have done our best to ensure the comparability to the challenge evaluation through communication with the challenge organizers and by successfully reproducing the scores for other submissions. For this experiment, in contrast to the previous leave-1-out setting, we train on the 15 training atlases, and perform the evaluation on the 20 testing target images. We compute the brain masks for this dataset with the parameterless ROBEX tool (Iglesias et al., 2011b)[4]. With the above settings, our mean Dice is 72.75 ± 7.03% over all labels (69.91 ± 7.44% for cortical, 80.49 ± 5.91% for non-cortical structures) with a running time of ca. 2 min for testing with atlas forests. In Fig. 8, we place our results in the context of the 25 challenge submissions. Overall, we observe accuracy corresponding closely to the mean and median of other approaches, with slightly below-average performance on cortical structures, and slightly above-average performance on non-cortical structures.

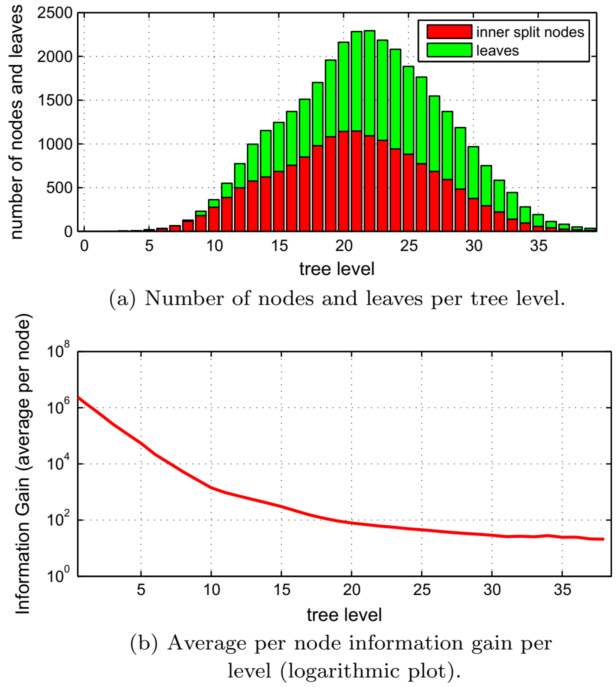
3.4. MICCAI 2013 SATA Challenge

The last experiment is performed on the unregistered version of the diencephalon data set from the MICCAI 2013 Challenge

[3] (Wu et al., 2012) does not state which 2 labels are omitted, we assume these are also the cerebellum and the brainstem.

[4] Available from http://www.nitrc.org/projects/robex.

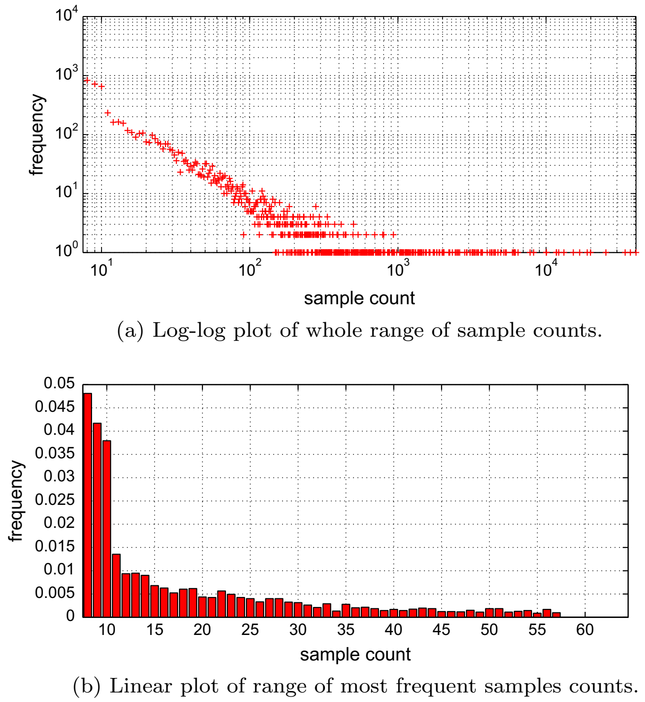
Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010



(a) Number of nodes and leaves per tree level.

(b) Average per node information gain per level (logarithmic plot).

Fig. 9. Tree analysis: (a) Distribution of inner nodes and leaves of the tree over levels and (b) the corresponding average information gain per inner node over levels.



(a) Log-log plot of whole range of sample counts.

(b) Linear plot of range of most frequent samples counts.

Fig. 10. Tree analysis: statistics of sample counts per leaf.

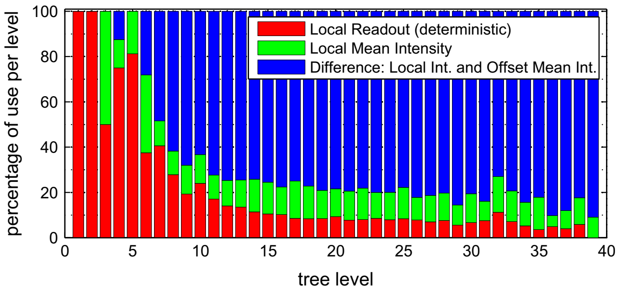
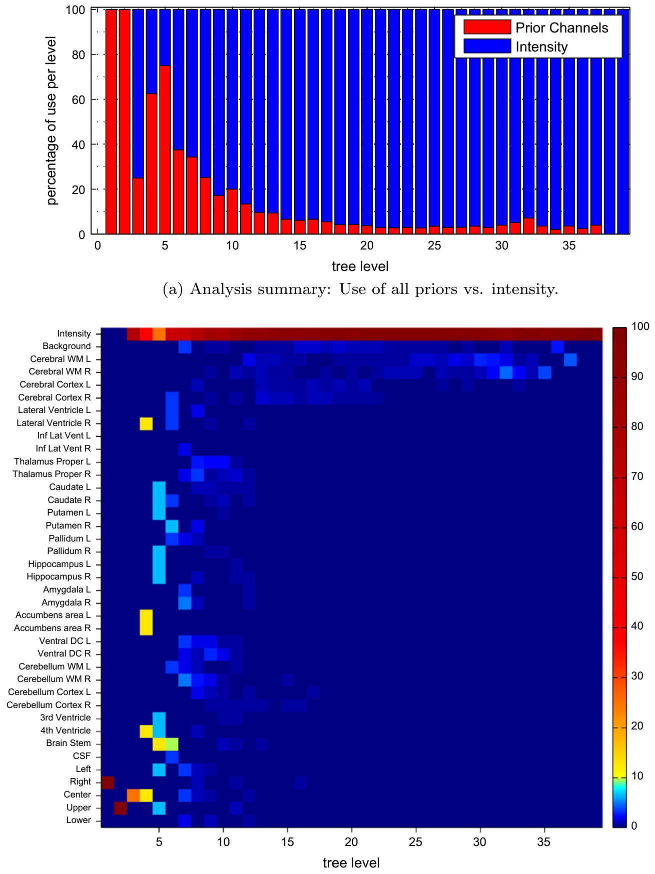


Fig. 11. Tree analysis: use of feature types per tree level.



(a) Analysis summary: Use of all priors vs. intensity.

(b) Detailed analysis: Please note the modification of the jet color map to enhance the visibility of small percentages.

Fig. 12. Tree analysis: use of channels per tree level.

Workshop on Segmentation: Algorithms, Theory and Applications (SATA) (Asman et al., 2013). The data consists of 35 training and 12 test T1 MR images from the OASIS project with corresponding 14 sub-cortical label maps as provided by Neuromorphometrics, Inc. under academic subscription. For this dataset, we compute the brain masks again with the ROBEX tool (Iglesias et al., 2011b). The evaluation is performed remotely by submitting to the challenge evaluation system. We obtain a Dice score of 82.47 ± 4.44% and a Hausdorff distance of 3.84 ± 0.73 mm. The time for applying the atlas forests to a single target image is ca. 2 min. Fig. 7d shows the leave-1-out cross-validation results on the training data.

3.5. Tree analysis

The performance behavior of our method is largely determined by the trees which are the result of the training process. Therefore, we try to summarize the properties of the tree structure and the node statistics in this section which hopefully provides further insights to our method.

We perform the analysis on a typical tree which was trained as part of the experiment on the IBSR dataset (max. depth = 40, min. samples = 8). This tree has 34,387 nodes, of which there are 17,193

Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010

inner nodes and 17,194 leaves. The atlas on which the tree was trained provides 1,040,178 samples. In Fig. 9a, we can see the distribution over inner nodes and leaves over the levels of the tree and observe that the chosen depth does not significantly limit tree growth – at this point the tree training basically runs out of samples.

In Fig. 9b we show the corresponding average information gain per inner node per level (on a logarithmic scale). The information gain per node becomes very small at deeper levels of the tree.

When it comes to the actual number of samples per leaf, it can be seen in Fig. 10 that the "small" leaves with very small sample counts are the most frequent. Very few "large" leaves are contained in the tree.

Finally, we analyze which feature types and channels get used in the tree by computing the usage percentage per level. In Fig. 11, we can see that the deterministic local readout feature dominates the first few tree levels, and that after that the difference feature becomes dominant, while the local mean box readout has approximately constant importance across the levels. When analyzing channel use in Fig. 12, one can see that the very top levels are dominated by the prior channels from the probabilistic atlas, and that on the lower levels the intensity is the main source of information. Among the prior channels, the aggregate priors are used before the regular single-label priors. An interpretation of these observations is that the algorithm uses the prior channels at the top levels to partition the samples into spatial subregions, and then primarily intensity-driven discrimination is learned for these regions. Because the features used on the prior channels are deterministic (available during training at each node), the structure of the top levels of the trees is very stable for all the atlases.

4. Discussion and summary

When comparing the proposed method to standard forest schemes, two interesting points arise: relation of our approach to standard bagging strategies, and the issue of over-training.

Bagging is a strategy for diversifying trees through randomization, by selecting a random subset of samples for the training of each tree. Single trees are then non-linear probabilistic approximating functions for a random sample subset, and the forest prediction is their linear combination. This strategy has the effect of improving generalization (Breiman, 2001). Standard bagging strategies pool samples for each tree indiscriminately from all available datasets (i.e. atlases in our application). A possible interpretation of our approach is to consider it as a specific bagging strategy, where the samples are not randomly chosen for each tree, but originate deterministically from a specific atlas. While such an approach can be expected to generalize poorly for general applications, our experiments in Section 3.1.2 show that this specific bagging strategy achieves similar accuracy levels in the studied settings. A potential explanation for this observation is that this is a property of the brain labeling application: Due to the similarity of the brain images, drawing samples from a single image or a set of different images can be expected to result in a similar distribution. If this assumption is not met, we would expect to see a decrease in the performance of the proposed scheme. For example, one issue that our current implementation might face would be a strong variation in scale, since we do not perform any explicit steps to deal with this issue to which the learned non-local features might be sensitive.[5]

Over-training is an important issue for learning-based algorithms. One interesting aspect of our method is that the used setting (trees with large depth and small number of samples per leaf) can be considered to lead to over-training, and accordingly, we observe a much higher accuracy of a single atlas forest on the corresponding atlas image, than the accuracy on the test images, cf. Fig. 4. However, our experiments on the variation of these parameters in Section 3.1.4 show that these settings ultimately – after the fusion step – do lead to improved performance compared to more conservative ones. A possible explanation for this observation is that we basically use the classifier as an encoding of an atlas, inside the MALP scenario. In this capacity, its ability to represent the atlas to a high degree (i.e. to over-train to the atlas) can be seen as an approximation to the standard MALP scheme with standard (i.e. no explicit) encoding of the atlas as an image/label-map pair.

In summary, in this work we propose to encode an atlas consisting of an intensity image and a corresponding label map by training a classifier exclusively on samples from that atlas. As a classifier, we use randomized forests because of their efficiency at test time and inherent capability for efficient multi-label classification. Compared to multi-atlas label propagation methods, our atlas encoding differs from the currently standard representations as an image/label-map pair, or a set of local patch collections. Also, while previous methods use a static encoding for all points in the image domain, our approach learns a flexible representation depending on the local context of the individual points. Compared to standard learning schemes, which pool samples indiscriminately across all atlases, our approach has a number of advantages for the MALP setting while preserving accuracy, such as the ability for atlas selection and addition of new atlases.

In terms of overall accuracy, our implementation of the proposed method shows performance corresponding roughly to the average of current methods, with some state of the art methods showing a clearly higher accuracy (compare Fig. 8). Possible steps to improve the accuracy are use of better registration, improved features, more sophisticated fusion, and further tuning to respective data sets.

The major practical advantage of our approach compared to existing MALP methods is the high efficiency. This is based on the inherent efficiency of our tree-based encoding, and the fact that only a single registration is required to label a target image. In return, compared to previous approaches, our method requires a training stage and the availability or creation of a probabilistic atlas. Overall, our approach achieves accuracy within the range of the state of the art, however at a much lower computational cost, both for the actual use of the system for labeling, as well as for experimentation.

References

Akhondi-Asl, A., Warfield, S., 2013. Simultaneous truth and performance level estimation through fusion of probabilistic segmentations. IEEE TMI.

Aljabar, P., Heckemann, R., Hammers, A., Hajnal, J., Rueckert, D., 2009. Multi-atlas based segmentation of brain images: atlas selection and its effect on accuracy. Neuroimage 46 (3), 726–738.

Asman, A., Landman, B., 2012a. Multi-atlas segmentation using spatial STAPLE. In: MICCAI Workshop on Multi-Atlas Labeling.

Asman, A.J., Landman, B.A., 2012b. Multi-atlas segmentation using non-local STAPLE. In: MICCAI Workshop on Multi-Atlas Labeling.

Asman, A., Akhondi-Asl, A., Wang, H., Tustison, N., Avants, B., Warfield, S.K., Landman, B., 2013. Miccai 2013 segmentation algorithms, theory and applications (sata) challenge results summary. In: MICCAI Challenge Workshop on Segmentation: Algorithms, Theory and Applications (SATA).

Breiman, L., 2001. Random forests. Machine Learning.

Coupé, P., Manjón, J., Fonov, V., Pruessner, J., Robles, M., Collins, D., 2011. Patch-based segmentation using expert priors: application to hippocampus and ventricle segmentation. Neuroimage 54 (2), 940–954.

Criminisi, A., Shotton, J. (Eds.), 2013. Decision Forests for Computer Vision and Medical Image Analysis. Springer.

Glocker, B., Komodakis, N., Tziritas, G., Navab, N., Paragios, N., 2008. Dense image registration through MRFs and efficient linear programming. MedIA.

[5] A possible remedy would be to present each AF with differently scaled versions of the atlas during training.

Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010

Heckemann, R., Hajnal, J., Aljabar, P., Rueckert, D., Hammers, A., et al., 2006. Automatic anatomical brain MRI segmentation combining label propagation and decision fusion. Neuroimage 33 (1), 115–126.

Iglesias, J.E., Konukoglu, E., Montillo, A., Tu, Z., Criminisi, A., 2011a. Combining generative and discriminative models for semantic segmentation of CT scans via active learning. In: IPMI.

Iglesias, J.E., Liu, C.Y., Thompson, P., Tu, Z., 2011b. Robust brain extraction across datasets and comparison with publicly available methods. IEEE TMI 30 (9), 1617–1634.

Joshi, S., Davis, B., Jomier, M., Gerig, G., 2004. Unbiased diffeomorphic atlas construction for computational anatomy. Neuroimage 23, S151–S160.

Klein, A., Tourville, J., 2012. 101 Labeled brain images and a consistent human cortical labeling protocol. Front. Brain Imag. Methods.

Klein, A., Andersson, J., Ardekani, B.A., Ashburner, J., Avants, B., Chiang, M.-C., Christensen, G.E., Collins, D.L., Gee, J., Hellier, P., et al., 2009. Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. Neuroimage 46 (3), 786–802.

Landman, B., Warfield, S. (Eds.), 2012. MICCAI 2012 Workshop on Multi-Atlas Labeling.

Montillo, A., Shotton, J., Winn, J., Iglesias, J.E., Metaxas, D., Criminisi, A., 2011. Entangled decision forests and their application for semantic segmentation of CT images. In: IPMI.

Rohlfing, T., Brandt, R., Menzel, R., Maurer, C., 2004. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. Neuroimage 21 (4), 1428–1442.

Rohlfing, T., Brandt, R., Menzel, R., Russakoff, D.B., Maurer Jr, C.R., 2005. Quo vadis, atlas-based segmentation? In: Handbook of Biomedical Image Analysis. Springer, pp. 435–486.

Rohlfing, T., Zahr, N.M., Sullivan, E.V., Pfefferbaum, A., 2010. The SRI24 multichannel atlas of normal adult human brain structure. Hum. Brain Mapp..

Rousseau, F., Habas, P., Studholme, C., 2011. A supervised patch-based approach for human brain labeling. IEEE TMI 30 (10), 1852–1862.

Shattuck, D., Mirza, M., Adisetiyo, V., Hojatkashani, C., Salamon, G., Narr, K., Poldrack, R., Bilder, R., Toga, A., 2007. Construction of a 3d probabilistic atlas of human cortical structures. Neuroimage 39 (3), 1064–1080.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A., 2011. Real-time human pose recognition in parts from single depth images. In: IEEE Computer Vision and Pattern Recognition (CVPR).

Tu, Z., 2005. Probabilistic boosting-tree: learning discriminative models for classification, recognition, and clustering;;. In: ICCV 2005. Tenth IEEE International Conference onComputer Vision, 2005, vol. 2. IEEE, pp. 1589–1596.

Tu, Z., Bai, X., 2010. Auto-context and its application to high-level vision tasks and 3D brain image segmentation. PAMI 32 (10), 1744–1757.

Tu, Z., Narr, K.L., Dollar, P., Dinov, I., Thompson, P.M., Toga, A.W., 2008. Brain anatomical structure segmentation by hybrid discriminative/generative models. IEEE Trans. Med. Imag. 27 (4).

Tustison, N., Gee, J., 2010. N4ITK: Nick's N3 ITK implementation for MRI bias field correction. Insight J..

Wang, H., Das, S.R., Suh, J.W., Altinay, M., Pluta, J., Craige, C., Avants, B., Yushkevich, P.A., 2011. A learning-based wrapper method to correct systematic errors in automatic image segmentation: consistently improved performance in hippocampus, cortex and brain segmentation. Neuroimage 55 (3), 968–985.

Wang, H., Avants, B., Yushkevich, P., 2012. A combined joint label fusion and corrective learning approach. In: MICCAI Workshop on Multi-Atlas Labeling.

Wang, Z., Wolz, R., Tong, T., Rueckert, D., 2013. Spatially aware patch-based segmentation (saps): an alternative patch-based segmentation framework. In: Menze, B.H., Langs, G., Lu, L., Montillo, A., Tu, Z., Criminisi, A. (Eds.), Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging, Lecture Notes in Computer Science, vol. 7766. Springer, Heidelberg, pp. 93–103.

Warfield, S., Zou, K., Wells, W., 2004. Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation. IEEE TMI 23 (7), 903–921.

Wu, G., Wang, Q., Zhang, D., Shen, D., 2012. Robust patch-based multi-atlas labeling by joint sparsity regularization. In: MICCAI Workshop STMI.

Zikic, D., Glocker, B., Konukoglu, E., Criminisi, A., Shotton, J., Demiralp, C., Thomas, O., Das, T., Jena, R., Price, S., 2012. Decision forests for tissue-specific segmentation of high-grade gliomas in multi-channel MR. In: MICCAI.

Zikic, D., Glocker, B., Criminisi, A., 2013a. Atlas encoding by randomized forests for efficient label propagation. In: MICCAI.

Zikic, D., Glocker, B., Criminisi, A., 2013b. Multi-atlas label propagation with atlas encoding by randomized forests. In: MICCAI 2013 Challenge Workshop on Segmentation: Algorithms, Theory and Applications (SATA).

Please cite this article in press as: Zikic, D., et al. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Med. Image Anal. (2014), http://dx.doi.org/10.1016/j.media.2014.06.010