Atlas Encoding by Randomized Forests for Efficient Label Propagation

Darko Zikic, Ben Glocker, and Antonio Criminisi

Microsoft Research Cambridge

Abstract We propose a method for multi-atlas label propagation 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 negatively affects 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). At test time, each AF yields a probabilistic label estimate, and 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, incorporation of new scans is possible without retraining, and target-specific selection of atlases remains possible. The evaluation on three different databases shows accuracy at the level of the state of the art, at a significantly lower runtime.

1 Introduction

Labeling of healthy human brain anatomy is a crucial prerequisite for many clinical and research applications. Due to the effort involved in fully manual labeling 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 [1] 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. 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. [2,3,4]. The second group of patch-based methods has recently enjoyed increased attention [5,6,7]. 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 [5,6], resulting in labeling runtimes of 22-130 minutes per target on the IBSR dataset [6]. However, note that these approaches do not change the actual number of required registrations. The fusion step, which combines the atlas-specific label proposals into a final estimate, aims to correct for inaccurate registration or labellings, and remains an active research topic.

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 runtimes for the best performing registration schemes for the problem [8]. Current methods state runtimes of 2-20 hours per single registration [1]. Furthermore, sophisticated fusion schemes can also be computationally expensive. State of the art approaches state fusion runtimes of 3-5 hours [9,10,11] on a database of 15 atlases [1]. 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 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 [7], or use of label-specific kNN search structures [12]. 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 performed by averaging of the probability estimates. 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 runtime is independent of the database size in this respect.
2. *Efficient generation of atlas proposals and their fusion.* For proposal generation one AF per atlas is evaluated, and the fusion consists is done by averaging. While both operations scale linearly with database size, they are significantly more efficient than current approaches. For example, for the database with 15 atlases from [1], labeling of a single target takes ca. 4 min.
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 a given probabilistic atlas which is not part of experimentation).

Besides being efficient, experiments on 3 databases in Sec. 3 indicate that our scheme also achieves high accuracy, comparing favorably to state of the art.

Compared to standard forest schemes (cf. e.g. [13,14,15,16]) which train each tree on data from *all* training images, our model, which trains each tree on a single atlas exemplar, has three advantageous properties for MALP.

1. *Simple incorporation of new atlases into the database.* For standard forest schemes, non-approximative addition of new training data requires complete retraining. 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 step seems non-obvious for standard forest schemes. This property allows use of atlas-selection [17], which can reduce the computational cost, improve accuracy.
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.

After presenting the method in Sec. 2, and demonstrating its performance in Sec. 3, we summarize and discuss its properties in Sec. 4.

2 Method - Atlas Forests

An atlas forest (AF) encodes a single atlas by training a classification forest exclusively on the data from the atlas. AFs do not depend on the reference frame of the target image, since every point is described only by its appearance, without considering its location (this can be seen as a further relaxation of the one-to-one assumption). While this allows us to avoid the registration of atlases to the target, a problem with such a location-oblivious approach is that the location carries valuable information about label probabilities (e.g. a point on the far left is unlikely to carry a right-side label). To efficiently integrate this information, we augment the intensity information from the atlas/target image by label priors warped to the image, and AFs operate on this augmented input. For the alignment of the priors, only a *single* registration per image is required.

2.1 Forest Training, and Labeling by Testing and Fusion

We use randomized forests as a classifier since they can efficiently handle a high number of classes, which is important in the MALP setting. Since we use a standard forest type, we keep the description short, and refer for details to e.g. [18,19]. Classification forests consist of a set of trees, and as a supervised learning method, they operate in two stages: training and testing. During training, each tree of the AF a_i is trained on the specific atlas image I_i and the corresponding label map L_i which contains label class values c. Specifically, each tree t learns a label class predictor $p_t(c|f)$ for a high-dimensional feature representation f of

points from I_i. The training is performed by splitting the training examples at each node based on their representation in the feature space. The split functions are computed by maximizing the *information gain* in randomly selected dimensions of the feature space. In this work, we stop the tree growth at a certain tree depth ($d=36$), with the condition that no tree leaf contains less than a certain number of samples ($s_{\min}=8$). Since we are dealing with a high number of classes with extremely varying sizes, we use class re-weighting, i.e. we adjust the probability computation for each class according to its frequency, such as to obtain a uniform distribution at the root node. Without this standard step, small classes would have low influence, resulting in reduced accuracy for these classes. 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 samples.

At testing, a target image I is labeled by processing its points by 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 the single tree predictions $p_a(c|f) = \frac{1}{n_t}\sum_{i=1}^{n_t} p_{t_i}(c|f)$. The fusion of these probabilistic estimates from n_a AFs is done by averaging, i.e. $p(c|f) = \frac{1}{n_a}\sum_{i=1}^{n_a} p_{a_i}(c|f)$, and subsequent maximum selection $\hat{c} = \arg\max_c p(c|f)$.

2.2 Features and Label Priors

We describe the intensity around a certain location by a bank of generic intensity-based parametric features, which are non-local but short-range. Given the point of interest x in image I, offset vectors u,v, cuboids $C_s(x)$ (centered at x with side lengths s,r), and the mean operator μ, we use the following feature types:

1. Local cuboid mean intensity: $\mu(I(C_s(x)))$
2. Difference of local intensity and offset cuboid mean: $I(x) - \mu(I(C_s(x+u)))$
3. Difference of local and offset cuboid means: $\mu(I(C_s(x))) - \mu(I(C_s(x+u)))$
4. Difference of offset cuboid means: $\mu(I(C_s(x+u))) - \mu(I(C_r(x+v)))$

The feature type and the above parameters (u,v,s,r) are drawn randomly during training at each node, thus defining the random feature space dimensions to be explored. Guided by the results from patch-based works [5,6], we use a maximum offset of 10mm, and cuboid side length $s, r < 5$mm.

Additionally to the random features, we use a set of deterministic features, which are considered at every node. These features are the local intensity $\tilde{I}(x)$ in a multi-channel image $\tilde{I}$, which is formed by augmenting the atlas image I by the aligned label priors P_L. Next to the priors for the individual labels, we employ further 6 priors, which aggregate 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 $|L|+7$-dimensional 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. 1.

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. [20]. The construction is performed by iterative registration of the training images to their mean [21]. This results in an average intensity image $\bar{I}$, and applying the computed warps to corresponding label maps followed by averaging yields a set of label priors P_L. To account for potential registration errors at test time, we smooth the prior maps by a Gaussian with $\sigma = 2$mm. We use affine registration, followed by a deformable registration by the FFD-based method from [22], with cross-correlation as data term, and conservative deformable settings with an FFD-grid spacing of 20mm and strong regularization. The registration operates on images down-sampled by a factor of 4, taking less than 30 seconds 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.

3 Evaluation

We evaluate our approach on three brain MRI databases. For all tests we perform the standard preprocessing steps: skull-stripping (own implementation), inhomogeneity correction [23], and histogram matching (www.itk.org).

We used the IBSR dataset in this work for the development of the method and the parameter settings. The same settings were then used also for the evaluation on the other two databases. As final settings, we use 5 trees per atlas forest. The single trees are trained down to depth of 36, with the restriction that each leaf contains at least 8 samples. Each node uses 1000 features from a pool of 10000 random features per tree. The training of one tree takes on average ca. 36 minutes on a standard desktop PC (Intel Xeon E5520 2.27GHz, 12 GB RAM). The runtimes reported below are for the label propagation only, and do not include the time for the registration of the probabilistic atlas (ca. 30 seconds), and the preprocessing of the target image.

IBSR Database. The IBSR data (http://www.cma.mgh.harvard.edu/ibsr/) contains 18 labeled images with 32 labels. To provide a comparative context, we cite the results from [6], which are shown to compare favorably to average dice scores (DSC) reported previously for the IBSR data. The IBSR data set is used in [6] in a leave-one-out evaluation, and the best performing version of the proposed method (group-wise multipoint) reaches a mean DSC of 83.5%, with a runtime of 130 minutes. A different variant discussed in [6] (fast multipoint), which aims at faster runtimes by performing the search at a reduced number of locations in the image, reaches a DSC of 82.25%, with a labeling runtime of 22 minutes. Our approach with the above settings reaches a DSC of 84.60% with a runtime of 3 minutes per target image. Further, we quantify the influence of some elements of our method on IBSR data (all by leave-one-out experiments):

- Using the proposed AF scheme without the augmentation by label priors significantly reduces the DSC to 77.38%, and introduces noise and extreme

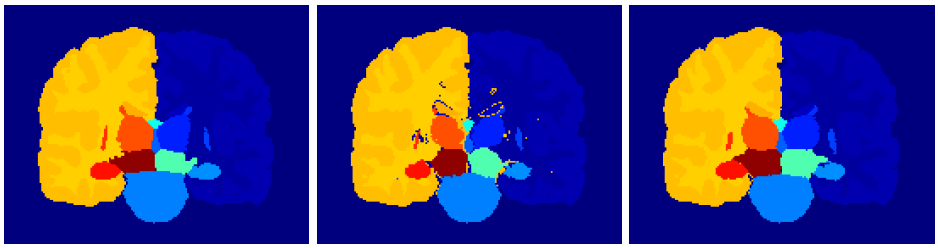
errors, as the forest is no longer able to compensate for the missing location information, see Fig. 1(b) for a visualization.
- Using only affine registration for construction and warping of the label priors decreases the DSC to 82.71%, indicating that accuracy improvement through a dedicated registration method might be possible [8].
- A standard forest scheme which uses approximately the same amount of data for training of each tree (i.e. 1/17th of all data), but randomly draws samples from all training images (i.e. performs bagging), reaches a DSC of 84.08%, with otherwise identical settings. This shows that our method does not reduce the quality, while introducing advantages for the MALP setting.

LONI-LPBA40 Database. The LONI-LPBA40 database [20] consists of 40 images of healthy volunteers, with 56 labels, most of them within the cortex. To provide some context, we cite the recent results on this data set from [7], where three methods are evaluated for 54 labels: an implementation of a patch-based scheme as in [5,6] (PBL), and two modifications aiming at sparsity of used patches (SPBL), and spatial consistency (SCPBL). The corresponding reported DSCs for a leave-one-out experiment are 75.06%, 76.46% and 78.04%, with runtimes of 10, 28 and 45 minutes *per class*. Our approach reaches an average DSC of 77.46% with a runtime of 8 minutes per image (for all classes).

MICCAI 2012 Multi-Atlas Labeling Challenge. Finally, we apply our approach to the data from [1], consisting of 15 training images and 20 test images from the OASIS project and corresponding label maps as provided by Neuromorphometrics, Inc. (http://Neuromorphometrics.com/) under academic subscription. The evaluation is performed on 134 labels (98 cortical, 36 non-cortical). Here, we train the AFs on the 15 training atlases, and perform the evaluation on the 20 testing target images. With the above settings, our mean DSC is 73.66% over all labels (71.04% for cortical, 80.81% for non-cortical structures) with a runtime of 4 minutes. In the evaluation in [1], this would place our approach in 8th position, out of 25 entries. The approach with the highest DSC in the challenge, PICSL-BC [9], reaches a score of 76.54%. A significant source of error in our approach seems to be a wrong labeling of background labels due to the used skull stripping – by restricting the evaluation to the reference brain masks, our approach would achieve 76.06%, while PICSL-BC would increase to 77.76%.

4 Summary and Discussion

We presented an efficient scheme for encoding of individual atlases for the purpose of multi-atlas label propagation. It represents an atlas by an atlas-specific classification forest, which is in contrast to the currently standard representations as an image/label-map pair, or a set of local patch collections. While previous methods use a static encoding for all points in the image domain, our approach learns a variable representation depending on the local context of the particular points. The major practical advantage of our approach is that only a single registration is required to label a target image. In return, compared to



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

Figure 1: Labeling example (IBSR): Using intensity-based features only (b) leads to extreme errors, which can be removed by additional use of label priors (c).

previous approaches, we require a training stage and a probabilistic atlas. However, we show that these additional requirements are not prohibitive. Compared to standard forest schemes, our approach has a number of advantages for label propagation, without loss of accuracy. Overall, our approach achieves accuracy comparable to state of the art at a much lower computational cost, both for the actual use of the system for labeling, as well as for experimentation.

With our approach in an early stage, we see several potential directions for improvement. Use of better atlases [20], registration [8], or skull-stripping might improve results. Early tests indicate that the size of the used feature space can be reduced without loss in accuracy, leading to more efficient training. Finally, adopting existing fusion approaches (e.g. [24]) is an interesting future direction.

References

1. Landman, B., Warfield, S., eds.: MICCAI Workshop on Multi-Atlas Labeling. (2012)
2. Rohlfing, T., Brandt, R., Menzel, R., Maurer, C.: Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. NeuroImage **21**(4) (2004) 1428–1442
3. Warfield, S., Zou, K., Wells, W.: Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation. IEEE TMI **23**(7) (2004) 903–921
4. Heckemann, R., Hajnal, J., Aljabar, P., Rueckert, D., Hammers, A., et al.: Automatic anatomical brain MRI segmentation combining label propagation and decision fusion. NeuroImage **33**(1) (2006) 115–126
5. Coupé, P., Manjón, J., Fonov, V., Pruessner, J., Robles, M., Collins, D.: Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. NeuroImage **54**(2) (2011) 940–954
6. Rousseau, F., Habas, P., Studholme, C.: A supervised patch-based approach for human brain labeling. IEEE TMI **30**(10) (2011) 1852–1862
7. Wu, G., Wang, Q., Zhang, D., Shen, D.: Robust patch-based multi-atlas labeling by joint sparsity regularization. In: MICCAI Workshop STMI. (2012)

8. 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.: Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. Neuroimage **46**(3) (2009) 786–802
9. Wang, H., Avants, B., Yushkevich, P.: A combined joint label fusion and corrective learning approach. In: MICCAI Workshop on Multi-Atlas Labeling. (2012)
10. Asman, A.J., Landman, B.A.: Multi-atlas segmentation using non-local STAPLE. In: MICCAI Workshop on Multi-Atlas Labeling. (2012)
11. Asman, A., Landman, B.: Multi-atlas segmentation using spatial STAPLE. In: MICCAI Workshop on Multi-Atlas Labeling. (2012)
12. Wang, Z., Wolz, R., Tong, T., Rueckert, D.: 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. Volume 7766 of LNCS. Springer (2013) 93–103
13. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: IEEE Computer Vision and Pattern Recognition (CVPR). (2011)
14. Iglesias, J.E., Konukoglu, E., Montillo, A., Tu, Z., Criminisi, A.: Combining generative and discriminative models for semantic segmentation of CT scans via active learning. In Székely, G., Hahn, H., eds.: Information Processing in Medical Imaging. Volume 6801 of LNCS. Springer (2011) 25–36
15. Montillo, A., Shotton, J., Winn, J., Iglesias, J.E., Metaxas, D., Criminisi, A.: Entangled decision forests and their application for semantic segmentation of CT images. In Székely, G., Hahn, H., eds.: Information Processing in Medical Imaging. Volume 6801 of LNCS. Springer (2011) 184–196
16. Zikic, D., Glocker, B., Konukoglu, E., Criminisi, A., Shotton, J., Demiralp, C., Thomas, O., Das, T., Jena, R., Price, S.: Decision forests for tissue-specific segmentation of high-grade gliomas in multi-channel MR. In Ayache, N., Delingette, H., Golland, P., Mori, K., eds.: Medical Image Computing and Computer-Assisted Intervention (MICCAI). Volume 7512 of LNCS. Springer (2012) 369–376
17. Aljabar, P., Heckemann, R., Hammers, A., Hajnal, J., Rueckert, D.: Multi-atlas based segmentation of brain images: Atlas selection and its effect on accuracy. NeuroImage **46**(3) (2009) 726–738
18. Breiman, L.: Random forests. Machine Learning (2001)
19. Criminisi, A., Shotton, J., eds.: Decision Forests for Computer Vision and Medical Image Analysis. Springer (2013)
20. Shattuck, D., Mirza, M., Adisetiyo, V., Hojatkashani, C., Salamon, G., Narr, K., Poldrack, R., Bilder, R., Toga, A.: Construction of a 3d probabilistic atlas of human cortical structures. NeuroImage **39**(3) (2007) 1064–1080
21. Joshi, S., Davis, B., Jomier, M., Gerig, G.: Unbiased diffeomorphic atlas construction for computational anatomy. Neuroimage **23** (2004) S151–S160
22. Glocker, B., Komodakis, N., Tziritas, G., Navab, N., Paragios, N.: Dense image registration through MRFs and efficient linear programming. MedIA (2008)
23. Tustison, N., Gee, J.: N4ITK: Nick's N3 ITK implementation for MRI bias field correction. The Insight Journal (2010)
24. Ledig, C., Wolz, R., Aljabar, P., Lötjönen, J., Heckemann, R., Hammers, A., Rueckert, D.: Multi-class brain segmentation using atlas propagation and em-based refinement. In: IEEE ISBI. (2012)