Laplacian Forests: Semantic Image Segmentation by Guided Bagging

Herve Lombaert[1,2], Darko Zikic[2], Antonio Criminisi[2], and Nicholas Ayache[1]

[1] INRIA Sophia-Antipolis, Asclepios Team, France
[2] Microsoft Research, Cambridge, UK

Abstract. This paper presents a new, efficient and accurate technique for the semantic segmentation of medical images. The paper builds upon the successful random decision forests model and improves on it by modifying the way in which randomness is injected into the tree training process. The contribution of this paper is two-fold. First, we replace the conventional *bagging* procedure (the uniform sampling of training images) with a *guided* bagging approach, which exploits the inherent structure and organization of the training image set. This allows the creation of decision trees that are specialized to a specific sub-type of images in the training set. Second, the segmentation of a previously unseen image happens via selection and application of only the trees that are relevant to the given test image. Tree selection is done automatically, via the learned image embedding, with more precisely a Laplacian eigenmap. We, therefore, call the proposed approach Laplacian Forests. We validate Laplacian Forests on a dataset of 256, manually segmented 3D CT scans of patients showing high variability in scanning protocols, resolution, body shape and anomalies. Compared with conventional decision forests, Laplacian Forests yield both higher training efficiency, due to the local analysis of the training image space, as well as higher segmentation accuracy, due to the specialization of the forest to image sub-types.

1 Introduction

The accuracy of image segmentation is crucial in many medical imaging applications, affecting notably diagnostics and treatments. Popular approaches often require manual user interactions [1,2], however, automatic approaches are increasingly benefiting from the rising number of annotated data. Early proposals, such as template-based methods [3] or Active Shape Models [4], were exploiting annotated ground truth data, known as atlases, in order to label previously unseen images. These statistical atlases were built using registration of training images, which is a non-trivial problem. Recently, machine learning approaches [5,6] propose to capture correlations between image features and associated ground truth labels, and exploit these learned traits to segment previously unseen images. Among such approaches, Random Decision Forests (RF) [7,8] have shown impressive, fast and accurate segmentation of medical images, including brain [9,10,11], and heart [12] images. They notably derive their strength from *bagging* [7], a uniform sampling of training images, which adds randomness during training of the decision trees. Image segmentation is simply done by labeling pixels with the predictions of the decision trees.

We propose to improve accuracy of the decision trees by modifying the way in which randomness is injected in the training process. Motivated by the fact that i)

training should consider affinities between images, and ii) decision trees should have different influences with respect to a test image, we decide to differentiate images by exploiting an embedding of the training image set. A large training set may indeed contain images of various anatomical regions, e.g., cerebral, thoracic, and/or abdominal. Bagging of too diverse images, with high variability in shapes and anomalies, may confuse training, which in turn, would necessitate an increased number of images for training. This increases memory and computational burden. Ideally, separate forests would be trained on specific sub-type of images in the training set, e.g., one would not annotate a thoracic image using a forest trained on abdominal images. This is inline with the Conditional Regression Forest [13,14], which focuses training of facial/body features on separate subsets of head/body-pose images. Pushing this strategy further, a recent attempt at localizing training and reducing computation burden for large databases is Atlas Forest (AF) [10], where trees are trained on single images, and predictions are made by averaging the probability estimates of these single-image trees. Grouping heterogeneous images into sub-types is, however, not trivial. Medical images pose an even greater challenge as their quality and field of view may be varying with acquisition protocols, whereas their DICOM tags for image types are not necessarily available or reliable.

The contribution of this paper is two-fold. First, we replace the uniform bagging strategy of conventional forests with *guided* bagging. Training images are embedded in a reduced image space representation, which exploits the underlying structure and organization of image sets, more precisely, via a Laplacian eigenmap. Decision trees are subsequently trained on images within *specific* neighborhoods of the embedding. Image sampling is *guided* rather than uniformly randomized. Guided bagging could also be regarded as a generalization of the Atlas Forest framework, where training is extended beyond single-image trees and uses neighborhoods of images on the Laplacian eigenmap. Second, we select decision trees based on their relevance to a test image. The segmentation of a previously unseen image is made by weighting the contributions of each individual tree with relative distances, and descriptive statistics, between training and testing image representations in the embedded space. A forest is, therefore, built at test time, as opposed to training time, and utilizes only the *revelant* trees depending on where a test image lands on the Laplacian eigenmap. We thus name our method **Laplacian Forests**. Furthermore, guided bagging and tree weighting allow Laplacian Forests to scale well with large databases, since each new training image would naturally bring information to only the relevant trees, increasing consistency within a decision tree during training. Similarly, new test images would pull information from only relevant trees without sacrificing computation costs at test time. The next section describes the Laplacian Forests in more details, followed by a set of experiments that illustrates the fundamental differences of Laplacian Forests over conventional forests. Results show how our approach improves segmentation accuracy on a dataset of 256 CT images with high variability in shapes and anomalies.

2 Method – Laplacian Forest

We begin by briefly reviewing Random Decision Forests (RF), and extend the general framework to exploit the underlying structure of the image space.

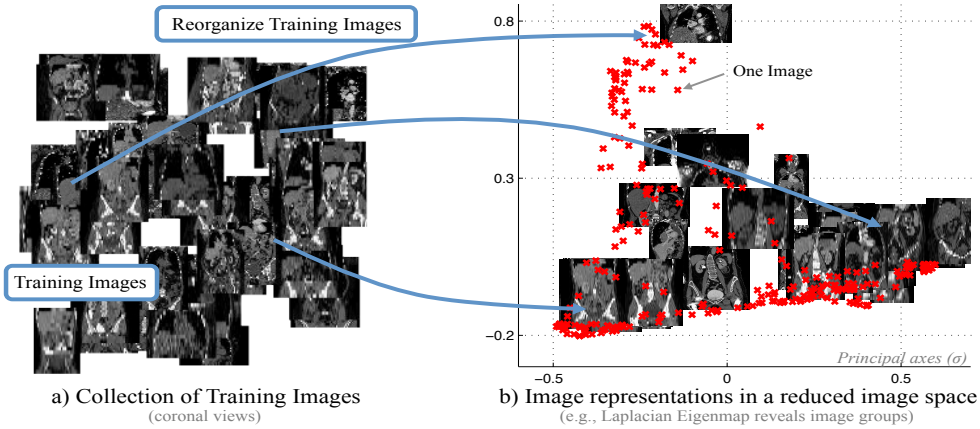


Fig. 1. Image Space Embedding – a) Collection of training images (ground truth), with no particular order. b) Reorganized images, ordered using the first two principal axes of a Laplacian eigenmap (images as red points). This map allows images to be grouped in neighborhoods, e.g., thoracic on top, abdominal in the bottom.

Standard Forest (RF) – A standard RF classifier consists of an ensemble of decision trees, each trained on a randomized subset of training data. During training, a tree is grown by establishing binary decisions that optimally split a training set from node to node such that *information gain* among class distributions is maximized. Each tree, t, learns a probabilistic class predictor $p_t(c|f)$ for a feature representation f of an image pixel. During testing, pixels of a test image are labeled by passing down their feature representations in n_{tree} trees. The resulting class predictions $\{p_{t_i}(c|f)\}_{1..n_{\text{tree}}}$ are averaged and a pixel is finally labeled using the maximal prediction $\hat{c} = \arg\max_c \sum_{i=1}^{n_T} p_{t_i}(c|f)$. More details could be found in [7,8]. A key aspect of RF is that accuracy and generalization are improved with *bagging*, a *uniform* sampling of the training image set. Atlas Forest (AF) [10] is a particular case of RF where individual trees are trained on single images. AF was shown to outperform the state-of-the-art in brain labeling [15].

Laplacian Forest (LF) – In LF, training images are reorganized into a new, low-dimensional continuous space. More precisely, they are embedded in a Laplacian eigenmap, as illustrated on Fig. 1. LF exploits such embedding in two novel ways: guided bagging during training, and tree weighting during testing. The general concept is i) to train trees on sets of similar images, which are in fact conveniently localized within neighborhoods of the reduced image space, and ii) to form forests at test-time with decision trees that are close to the mapping of a test image into the reduced image space. These strategies, illustrated on Fig. 2, contrast with standard uniform bagging, and with uniform tree weighting in RF.

2.1 Training Stage – Guided Bagging

Given a training set of n_{img} images $\{I_i\}_{1..n_{\text{img}}}$ with their associated label maps, affinities are first established between images, in order to build an embedding of the training images.

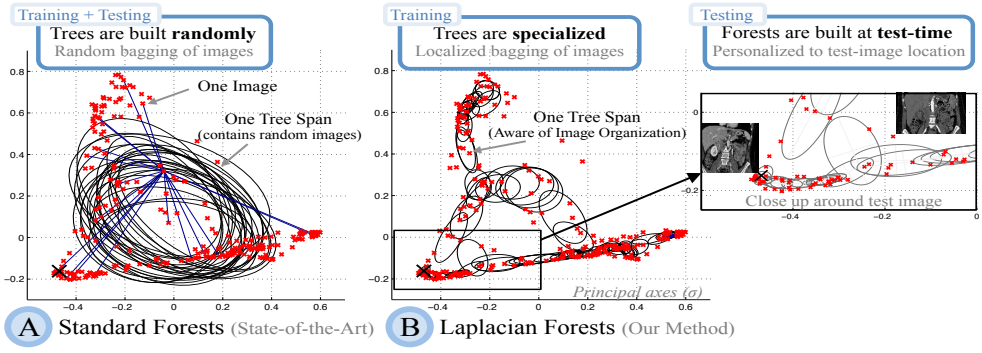


Fig. 2. Algorithm Comparison – a) Standard Forests use uniform bagging (training with randomly chosen images). One decision tree is shown by connecting in blue, images used for its training (covers large area with various image types). b) Laplacian Forests use guided bagging, by creating *specialized* trees with neighboring images on the Laplacian eigenmap. Trees are shown with ellipses that fit their constituting images. Testing only needs trees close to a test images (black cross, bottom left).

Affinity in Image Space – Image similarity is defined by a distance $d(i,j)$ between images I_i and I_j. In this paper, we use the sum of squared differences of pixel intensities between images: $d(i,j)^2 = \frac{1}{|I_i \cap I_j|} \sum_{p \in \{I_i \cap I_j\}} \left(I_i(p) - I_j(p)\right)^2$, which is fast and uses no image registration, nor any label information. If images have different sizes, we use their overlapping area $I_i \cap I_j$ around their center points. These distances are used to build the $n_{\text{img}} \times n_{\text{img}}$ *weighted adjacency* matrix W, defined in terms of image affinity, here: $w_{ij} = \exp\left(-d(i,j)^2/2\sigma^2\right)$ if images I_i and I_j are within their respective k-nearest neighbors, 0 otherwise, while σ represents tolerance to similarity, e.g., with the average of distances $\sigma = \text{mean}\left(d(i,j)\right)$. The diagonal *degree* matrix is the sum of all affinities $d_i = \sum_j w_{ij}$. Normalized affinities in the image space are summarized in the generalized *Laplacian* [16], a $n_{\text{img}} \times n_{\text{img}}$ matrix $L = D^{-1}(D - W)$.

Images on Laplacian Eigenmap – We reduce the high-dimensional image space via the spectral decomposition of the Laplacian $L = X \Lambda X^T$, giving the eigenvalues, in increasing order, $\Lambda = \text{diag}(\lambda_0, \lambda_1, ..., \lambda_{n_{\text{img}}})$ and their associated eigenvectors $X = (x^{(0)}, x^{(1)}, ..., x^{(n_{\text{img}})})$, a $n_{\text{img}} \times n_{\text{img}}$ matrix where each column $x^{(\cdot)}$ is an eigenvector. We finally define the m-dimensional spectral coordinate of an image I_i as $s_i = \left(\lambda_1^{-\frac{1}{2}} x^{(1)}(i), ..., \lambda_m^{-\frac{1}{2}} x^{(m)}(i)\right)$. We use in this paper $m = 2$. The Laplacian eigenmap, which is our embedding of the training image set, is the representation of images as points in m dimensions with normalized positions $\{s_i\}_{1..m}$, illustrated on Fig. 1b.

Laplacian Trees – We train n_{tree} trees on subsets of images that spreads across small neighborhoods on the Laplacian eigenmap. In order to evenly cover the reduced image space, we first find n_{tree} points by clustering all image spectral coordinates $\{s_i\}_{1..n_{\text{img}}}$ with k-means. The *Laplacian trees* are formed using each of these clusters. Their centers are defined as the average spectral coordinates of the images composing each cluster. Each of which is subsequently trained using $n_{\text{img/tree}}$ images chosen randomly within the vicinity of c_i on the Laplacian eigenmap, as illustrated on Fig. 2b. In addition to the center c_i of a tree t_i, we also retain the distribution of

the spectral coordinates, $S_i = (s_1; ...; s_{n_{\text{img/tree}}})$, of its constituting images via Principal Component Analysis, $S_i^T S_i = U_i \Sigma_i U_i^T$, where U_i contains the m major axes of the Laplacian tree, illustrated as ellipses on Fig. 2. A Laplacian tree differs from a conventional tree only because it is trained on images that are close to one another in the image space.

Training is done using the following commonly used features [9,10,11,12] for each pixel: a) the average intensity within a randomly-shaped cuboid centered around the tested pixel, b) the average intensity of a random cuboid centered around a randomly offset pixel, c) the differences of average intensities between two random cuboids centered around randomly offset pixels, d) the pixel coordinates (x, y, z).

2.2 Testing Stage – Tree Weighting

During testing, a test image I_t is first mapped on the Laplacian eigenmap. Its mapped position gives the neighboring Laplacian trees to be used for labeling the pixels of the test image.

Location of Test Images – Given a test image I_t, its k-nearest neighbor images (kNN) within the training set are found in terms of image similarity, i.e., we find the indices of the k training images with the smallest $d(i, t)$. The spectral coordinate of the test image is interpolated with $s_t = \frac{1}{Z} \sum_{i \in \text{kNN}(t)} w_{it} s_i$, where $Z = \sum_{i \in \text{kNN}(t)} w_{it}$. The position of a test image is illustrated with a black cross in Fig. 2b. Nearby Laplacian trees will have a higher influence for labeling the test image.

Weighting of Laplacian Trees – The relative influence between a test image and a Laplacian tree is measured with the Mahalanobis distance of the spectral coordinate of the test image, s_t, within the statistical spread of the Laplacian tree: $d^2_{\text{tree}}(t, i) = (s_t - c_i)\, U_i \Sigma^{-\frac{1}{2}}$. We further favor close trees by weighting them with: $w_{\text{tree}}(t, i) = \exp\left(-d^2_{\text{tree}}(t, i)/2\sigma^2\right)$, where the tolerance is $\sigma = \text{mean}\left(d^2_{\text{tree}}(t, i)\right)$.

Prediction – The final labeling of pixels in a test image I_t is done by taking the maximum weighted class predictions: $\hat{c} = \arg\max_c \left(\frac{1}{Z} \sum_{i=1}^{n_T} w_{\text{tree}}(t, i) p_{t_i}(c|f)\right)$, where $Z = \sum_{i=1}^{n_T} w_{\text{tree}}(t, i)$. Decision trees that are close to the test image on the embedding have, therefore, a stronger influence in the final class prediction. Such tree weighting differs from conventional RF methods where averages of tree contributions are used instead.

3 Results

We validate our method using a dataset of 256 3D CT images, manually labeled with 11 organs, all acquired at different hospital sites with varying acquisition protocols: volumes with 13 to 515 slices of size 128^2, resolutions ranging from 0.89 to 4.69mm, with and without contrast agent, in the thorax and abdomen (Fig. 1). We illustrate the fundamental differences between standard Random Decision Forests (RF) and Laplacian Forests (LF), which essentially reside in i) how images are selected via guided bagging during training, and ii) how decision trees are weighted during testing. Our experiments also avoid using any pre- or post-processing step in order to evaluate the direct improvement of the guided bagging and tree weighting strategies.

Bagging Strategy – We form a training set of 255 images, and use the remainder image as a test image. Accuracy is measured with the Dice metric $(2|A \cap B|/(|A| + |B|)$,

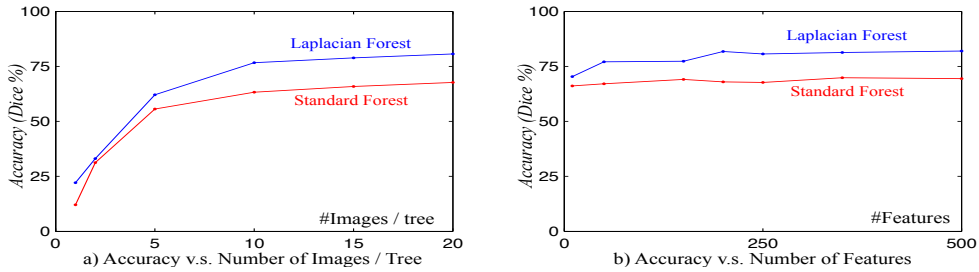


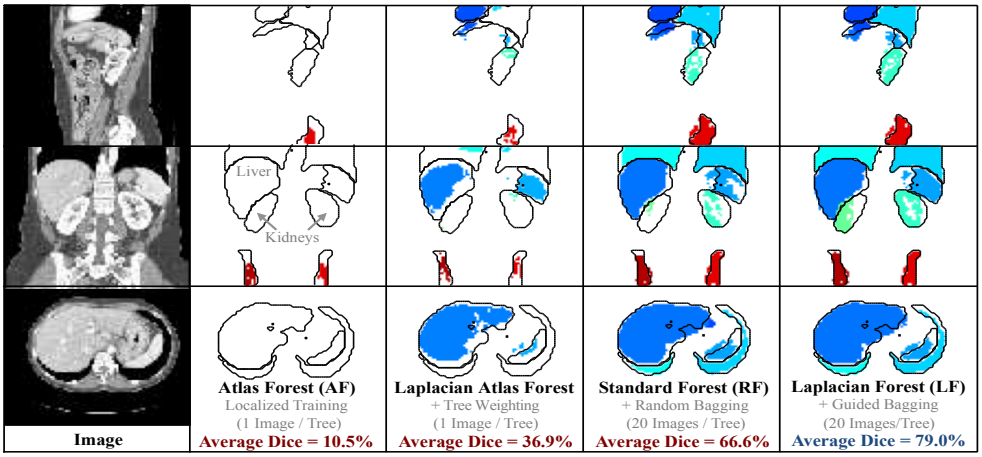
Fig. 3. Comparing bagging strategy – Standard Forests (red) and Laplacian Forests (blue). We increase a) the number of images per tree, and b) the number of features used for training. Experiments are repeated 10 times, average accuracy is measured with Dice metric, in %. Our guided bagging strategy improves accuracy by 18% when using 20 images per tree and 250 features.

where A is a ground truth segmentation for an organ, and B is our prediction). We use 50 trees, each trained using 15 levels, on 1 million pixels drawn randomly from a subset of $n_{\text{img/tree}}$ training images, where n_{features} features are computed as described earlier. We repeat each experiment 10 times and average the Dice scores for all organs using two settings: a) $n_{\text{img/tree}}$ is increased from 1 to 20 images per tree, and b) n_{features} from 10 to 500 features per tree. We observe in Fig. 3, that our bagging strategy (LF) increases the segmentation accuracy from 22% to 81%, while the conventional approach (RF) shows a lower increase from 12% to 68%. This shows that LF is more efficient in extracting discriminative information during training, perhaps, due to the use of more consistent decision trees. We further observe that increasing the cluster size, $n_{\text{img/tree}}$, has a stronger impact than increasing the number of features, n_{features}, which appears to be relatively stable from 50 to 500 features. This again highlights the impact of using localized decision trees.

Tree Weighting – We now evaluate the effect of weighting trees in the final labeling decision. Atlas Forest (AF) [10] is an extreme case of localized training where trees are trained on single images. We train 255 trees, each with $n_{\text{img/tree}} = 1$, $n_{\text{features}} = 250$, and test using the remaining image of the dataset. AF was shown to work well for brain labeling, where images exhibit relatively small variations. It is, however, not expected to handle highly heterogeneous databases as the one used here. Fig. 4 confirms indeed that segmentation fails for most organs and yields an average Dice score of 10.5%. We now modify AF to use weighted trees at test-time, which is in fact a special case of LF with $n_{\text{img/tree}} = 1$. Such, so-called Laplacian AF only adds a tree weighting strategy when compared to AF, and is shown to improve segmentation accuracy from 10.5% to 36.9%.

Guided Bagging + Tree Weighting – The previous experiment is now ran with $n_{\text{img/tree}} = 20$ using RF/LF. They use respectively uniform/guided bagging, without/with tree weighting. Combining both strategies improves the overall Dice performance from 66.6% (RF) to 79.0% (LF). One notable improvement is seen in smaller organs with high shape variability, such as in kidneys (green on Fig. 4).

Segmentation Accuracy – Finally, we cross-validate our dataset by using 9 tenth of images as training set, leaving 1 tenth as testing images. We repeat 10 times our validation, such that each image in the dataset is tested at least once. Table 1 shows



	Dice			
	AF	LAF	RF	LF
	1 img/cluster		20 img/cluster	
Heart	0%	76.3%	72.4%	74.5%
Liver	0%	74.3%	84.6%	86.9%
Spleen	0%	61.5%	52.4%	75.9%
Lung (L)	0%	3.5%	88.4%	88.5%
Lung (R)	0%	9.4%	88.4%	91.9%
Kidney (L)	0%	17.9%	41.8%	72.4%
Kidney (R)	0%	0.3%	5.8%	46.2%
Ilium (L)	48.3%	38.7%	84.2%	87.7%
Ilium (R)	46.3%	50.4%	81.3%	87.4%
Average	**10.5%**	**36.9%**	**66.6%**	**79.0%**

Fig. 4. Segmentation – Testing an image using an Atlas Forest (AF), Laplacian Atlas Forest (LAF) (an AF with tree weighting at test-time), Standard Forest (RF) and a Laplacian Forest (LF). Ground truth is overlaid in black contours for each organ. LF shows an overall improvement over RF, with a Dice metric increase from 66.6% to 79.0%. Large gains are noticeable in smaller organs such as in kidneys.

that on average, LF performs with a Dice score of 70.9%, which is a 5% increase over the standard RF. A similar improvement is seen when measuring the Jaccard index ($|A \cap B|/|A \cup B|$). Larger improvements are noticed in smaller organs, such as 30% increase in accuracy for kidneys.

4 Conclusion

We proposed two contributions for improving Random Decision Forests: i) guided bagging during training – decision trees are grown from subsets of images that are close to one another on a Laplacian eigenmap, and ii) non-uniform tree-weighting during testing – the contributions of individual decision trees are weighted based on their relative positions with respect to a test image on the Laplacian eigenmap. Laplacian Forests were shown to outperform standard forests in a dataset of CT

	Dice		Jaccard	
	RF	LF	RF	LF
Heart	68.2%	71.0%	54.6%	58.1%
Liver	82.7%	83.8%	71.4%	73.2%
Spleen	55.6%	57.8%	42.3%	44.5%
Lung (L)	91.7%	91.0%	86.2%	85.3%
Lung (R)	93.3%	93.1%	88.6%	88.4%
Kidney (L)	30.3%	39.8%	21.8%	29.4%
Kidney (R)	28.9%	37.7%	20.5%	28.1%
Femur (L)	75.4%	77.8%	63.7%	66.5%
Femur (R)	74.8%	78.1%	63.0%	66.8%
Ilium (L)	71.7%	74.5%	59.3%	62.3%
Ilium (R)	70.3%	74.3%	57.7%	62.0%
Average	**67.5%**	**70.9%**	**57.2%**	**60.4%**

Table 1. Cross-validation 10-folds – 256 CT Images – Standard Forests (RF) and Laplacian Forests (LF) – LF produces higher accuracy, notably for more challenging smaller organs, such as kidneys.

images of various anatomical regions. Larger improvements were noticeable in organs with higher variability of shape and positions, such as in kidneys.

Our *guided* bagging strategy produces decision trees with more consistent image information, since each tree is trained using related images. This could be seen as a generalization of Atlas Forests for using multiple images during training. Such localized information also has the advantage to be pooled efficiently during test-time. The final labeling decision is indeed based on weighted contributions from trees that are the most relevant to a test-image. Furthermore, our bagging strategy could scale well with additional training images, since information from new images would be simply exploited in only relevant trees. Our approach is also compatible with other variants of RF, such as for instance, Geodesic Forests [17], by simply changing the bagging strategy. In this paper, the Laplacian eigenmap was built using a very simple affinity measure, based on the sum of squared differences of images, yet, we observe an improvement in accuracy. Future work will focus on using better discriminating affinity measures as well as applying our strategy to other RF variants.

Acknowledgements – This research is partially funded by the ERC Advanced Grant MedYMA, Fonds de Recherche du Quebec (FRQNT), and the Research Council of Canada (NSERC).

References

1. Boykov, Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In: ICCV. (2001)[1]
2. Grady, L.: Random walks for image segmentation. PAMI **28**(11) (2006)[1]
3. Heimann, T., Meinzer, H.P.: Statistical shape models for 3D medical image segmentation: A review. MedIA **13**(4) (2009)[1]
4. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape Models-Their training and application. CVIU **61**(1) (1995)[1]
5. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2007)[1]
6. Wang, S., Summers, R.M.: Machine learning and radiology. MedIA **16**(5) (2012)[1]
7. Breiman, L.: Random forests. Mach. Learn. **45** (2001)[1, 3]
8. Criminisi, A., Shotton, J.: Decision Forests for Computer Vision and Medical Image Analysis. Springer (2013)[1, 3]
9. Zikic, D., Glocker, B., Konukoglu, E., Criminisi, A., Demiralp, C., Shotton, J., Thomas, O.M., Das, T., Jena, R., Price, S.J.: Decision forests for Tissue-Specific segmentation of High-Grade gliomas in multi-channel MR. In: MICCAI. (2012)[1, 5]
10. Zikic, D., Glocker, B., Criminisi, A.: Atlas encoding by randomized forests for efficient label propagation. In: MICCAI. (2013)[1, 2, 3, 5, 6]
11. Geremia, E., Menze, B., Clatz, O., Konukoglu, E., Criminisi, A., Ayache, N.: Spatial decision forests for MS lesion segmentation in multi-channel MR images. In: MICCAI. (2010)[1, 5]
12. Lempitsky, V., Verhoek, M., Noble, Blake, A.: Random forest classification for delineation of myocardium in Real-Time 3D echocardiography. In: FIMH. (2009)[1, 5]
13. Van Gool, L.: Real-time facial feature detection using conditional regression forests. In: CVPR. (2012)[2]
14. Sun, M., Kohli, P., Shotton, J.: Conditional regression forests for human pose estimation. In: CVPR. (2012)[2]
15. Rousseau, F., Habas, P.A., Studholme, C.: A supervised Patch-Based approach for human brain labeling. TMI **30**(10) (2011)[3]
16. Grady, L., Polimeni, J.R.: Discrete Calculus. Springer (2010)[4]
17. Kontschieder, P., Kohli, P., Shotton, J., Criminisi, A.: GeoF: Geodesic forests for learning coupled predictors. In: CVPR. (2013)[8]