Medical Image Analysis 23 (2015) 70–83

Contents lists available at ScienceDirect

Medical Image Analysis

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

Multi-organ localization with cascaded global-to-local regression and shape prior

Romane Gauriau [a,b,*], Rémi Cuingnet [a], David Lesage [a], Isabelle Bloch [b]

[a] Philips Research MediSys, 33 rue de Verdun, Suresnes Cedex 92156, France
[b] Institut Mines-Telecom, Telecom ParisTech, CNRS LTCI, 46 Rue Barrault, Paris 75013, France

ARTICLE INFO

Article history:
Received 21 October 2014
Revised 8 April 2015
Accepted 8 April 2015
Available online 17 April 2015

Keywords:
Multi-organ localization
Regression
Random forest
3D CT
Abdominal organs

ABSTRACT

We propose a method for fast, accurate and robust localization of several organs in medical images. We generalize the global-to-local cascade of regression random forest to multiple organs. A first regressor encodes the global relationships between organs, learning simultaneously all organs parameters. Then subsequent regressors refine the localization of each organ locally and independently for improved accuracy. By combining the regression vote distribution and the organ shape prior (through probabilistic atlas representation) we compute *confidence maps* that are organ-dedicated probability maps. They are used within the cascade itself, to better select the test voxels for the second set of regressors, and to provide richer information than the classical bounding boxes result thanks to the shape prior.

We propose an extensive study of the different learning and testing parameters, showing both their robustness to reasonable perturbations and their influence on the final algorithm accuracy. Finally we demonstrate the robustness and accuracy of our approach by evaluating the localization of six abdominal organs (liver, two kidneys, spleen, gallbladder and stomach) on a large and diverse database of 130 CT volumes. Moreover, the comparison of our results with two existing methods shows significant improvements brought by our approach and our deep understanding and optimization of the parameters.

1. Introduction

With the ever growing amount of 3D medical acquisitions, automatic, robust and accurate anatomy localization is of prime interest. First, it directly enables faster data navigation and visualization of target structures which can undoubtedly save the radiologist some time (Andriole et al., 2011). Second, organ localization is a key initialization step for tasks such as segmentation or registration (for the liver segmentation for example (Gauriau et al., 2013)). It is, overall, a crucial component to streamline complex workflows such as medical treatment planning and follow-up (in liver radiotherapy for instance, the volume of the liver is required to compute the dose (Murthy et al., 2005)). Such a component should be almost transparent in these kinds of applications. It should help the user or subsequent algorithms without a significant computation overhead and it should be accurate and robust enough to be reliable as an intermediary result. Therefore, the speed and the performance of the anatomy localization are two major aspects to take into account in the design of such an algorithm.

The method we propose follows these requirements and presents the advantage of being easily reproducible. The main contribution of this work is to provide a fast anatomy localization of multiple structures simultaneously with good accuracy and shape consistency in few seconds on a personal computer. This method is based on a cascade of regressions and the combination of localization predictions with organ dedicated probabilistic atlases. Fig. 1 presents the outline of the method applied to the localization of the right and left kidneys in a 3D CT image. The idea is first to predict the position of all the organs simultaneously with a single regressor and then to refine the localization organ per organ using dedicated regressors. This paper extends our preliminary work (Gauriau et al., 2014), in particular by giving a better theoretical description of the approach and proposing a detailed study and optimization of the learning and testing parameters.

In Section 1.1 we give a brief overview of the previous works on multi-organ localization and in Section 1.3 we summarize the contributions of this article. In Section 1.2 we give some background on the random forest method and existing extensions for object detection. Section 2 presents our approach and describes the principles behind confidence maps. Section 3 is dedicated to the study of an application of the proposed method, using the random forest as a regressor, for the localization of six abdominal organs: the liver, the two kidneys, the gallbladder, the spleen and the stomach. For that purpose we use a large and diverse database of 130 3D CT volumes. We

* Corresponding author.
E-mail address: romanegauriau@gmail.com (R. Gauriau).

http://dx.doi.org/10.1016/j.media.2015.04.007
1361-8415/

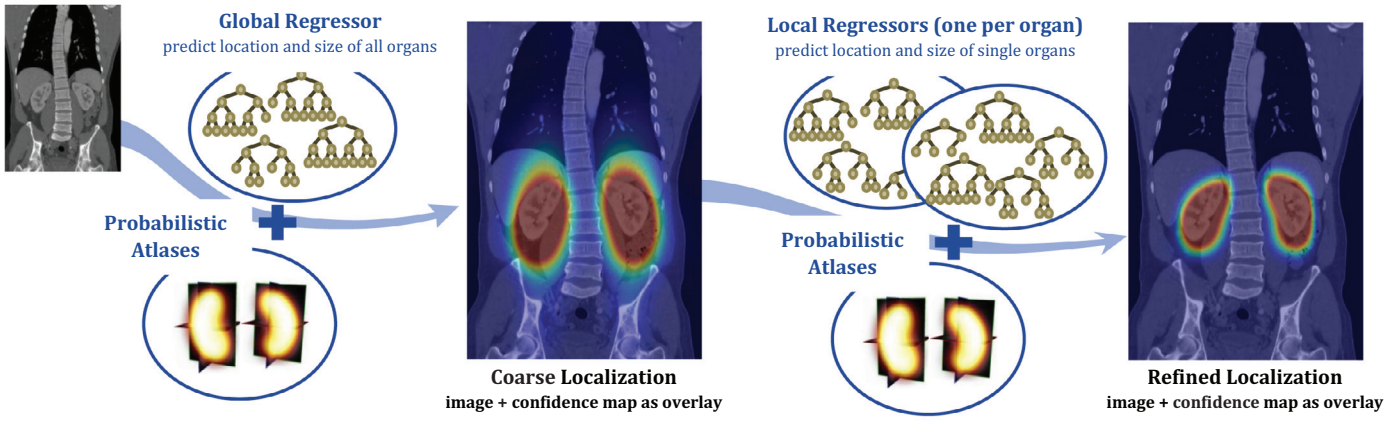

Fig. 1. Outline of the method applied to the localization of the kidneys in a 3D CT image. A cascade of random forest regressors provides a final accurate localization in the form of one confidence map per organ (both displayed as overlay over the image, red indicates a high chance of belonging to the organ while indicates a low chance). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

describe our implementation choices, so that the implementation can be reproduced by an interested reader. We also analyze the influence of a wide number of training and testing parameters on the overall performance of the method. Besides providing us with an optimal set of parameters, this study demonstrates our method's relatively low sensitivity to parameter variations. In Section 4 we evaluate our algorithm, in comparison with existing works. We illustrate the benefits of introducing organ-dedicated confidence maps.

1.1. Previous works on multi-organ localization

Localization aims at finding the position of a given structure while segmentation aims at determining the part of the image belonging to the said structure. Segmentation methods are often dependent on preliminary localization. While intrinsically linked, these two tasks often lead to vastly different formulations and algorithms. Numerous works deal with single organ localization. Among prominent ones, one can cite the marginal space learning (MSL) proposed by Zheng et al. (2007), already applied in several contexts (Seifert et al., 2010; Wu et al., 2014). Its goal is to reduce the localization search space by learning several detectors (classifiers) with increasing complexity (first find the best translated position in the image, then the best translated and rotated position, etc.). The generalization of such single organ methods to multiple organs localization is not always straightforward, except by doing it sequentially and thus not taking into account organ relationships. The importance of contextual information and spatial relationships has then led to specific methods for multi-structure localization. This is the reason why we focus this overview on such methods. We classify them into three groups of techniques: knowledge-based, atlas-based, and learning-based approaches.

Knowledge-based approaches. The spatial relationships of the anatomical structures can be represented by a graph where the nodes and the edges correspond to the organs and their relationships, respectively. Liu et al. (2011) used this representation in conjunction with organ-dedicated appearance models to label several organs. They solve this problem using dynamic programming. Unfortunately they do not give a quantitative evaluation or information about the computation time. Fouquier et al. (2012) proposed to localize and segment the brain structures in a pre-computed sequential order, following the prior knowledge of the spatial relationships between the different structures. The sequential nature of this process tends to propagate the errors and may require a backtracking procedure. To prevent such a problem, Nempont et al. (2013) proposed to solve the localization and segmentation in a global fashion following a constraint network. Although this approach has the benefit of relying on strong prior knowledge, it is computationally expensive in practice (several hours) and requires a very fine parameter tuning which makes it difficult to apply in practice until now.

Atlas-based approaches. Atlas-based approaches were originally used for segmentation purposes (see surveys on the topic such as Cabezas et al. 2011; Kalinic 2008). They are particularly computationally expensive. In the case of localization, the registration may be done at a lower resolution resulting in a reduced computational cost (Fenchel et al., 2008; Shimizu et al., 2005). Several approaches were proposed to handle the inter-patient variability by using a probabilistic atlas (Shimizu et al., 2005), a mean atlas (Joshi et al., 2004) or a multi-atlas approach (Blezek & Miller, 2007; Wolz et al., 2012). These approaches improve the standard atlas-based methods, albeit at the cost of increased computational time. In that respect, *Atlas Forests* (Zikic et al., 2013) propose an efficient, random forest-based alternative to the registration-based evaluation of textbook multi-atlas methods. Multi-atlas methods may still not be sufficient to cope with large variability, pathologies and inherent image-specific challenges. Recent developments such as the works of Lombaert et al. (2014) (*Laplacian Forests*) and Konukoglu et al. (2013) (*Neighborhood Approximation Forests*) exploit similarities between training images to capture more specific, more local variations (in the sense of a similarity metric) than classical global atlases/models. These works relate to multi-atlas approaches in that their key ideas can be used to build more specific, more accurate atlases while still benefiting from (local) clustering of training data. These works focus notably on learning and exploiting the appropriate similarity metrics within the framework of Random Forests (see also Section 1.2).

Learning-based methods. Machine learning techniques have recently raised high interest in the medical image processing community, probably benefiting from the computing power increase and from an easier access to large databases. A very good introduction to statistical learning can be found in Hastie (2009). The goal of statistical machine learning is roughly to model the dependence of an output variable y on an input variable x. This dependence can be defined as the posterior probability distribution $p(y|x)$ which helps predicting the value of an unobserved variable y. We distinguish two different ways of modeling $p(y|x)$: the generative model learns the joint distribution $p(y, x)$ while the discriminative model learns directly the conditional probability $p(y|x)$. Although some authors (Bishop & Lasserre, 2007; Ng & Jordan, 2001) showed the interest of generative models with the increase of sample numbers, discriminative models are still preferred in most of applications as the data true distribution may be difficult to estimate. If y corresponds to a label (or discrete variable), this problem is called *classification* (for

example $y = 1$ if x belongs to the target organ and $y = 0$ if it belongs to the background). If y corresponds to a continuous variable then the problem is called *regression*. The variable y may be a 3D position to predict for example. In image processing, classification-based methods often leverage the local context at the cost of an exhaustive scan of the image. In general, regression-based methods rely more on the global context but achieve faster speed for a lower accuracy (full parsing of the image is not required).

We distinguish two types of learning-based methods for multi-organ localization for which regression or classification may be used. The first one is based on landmarks detection, while the second one aims at finding regions of interest (ROI). Landmark-based methods are often used as a preprocessing step for positioning the organ models (Chen & Zheng, 2014; Seifert et al., 2009). The principle is to learn the position of anatomically meaningful and reproducible locations (e.g. the top of the right lung or the tip of the liver). Landmarks detection can be performed following different strategies. Landmarks can be detected independently (Chen & Zheng, 2014) or take into account the spatial relationships, for instance with a multivariate Gaussian model (Zhan et al., 2008) or with a graphical model (Seifert et al., 2009). Submodular strategies may also be designed to reduce the search range (Liu et al., 2010). Lay et al. (2013) proposed to increase the organ parameterization complexity by performing a joint anatomical landmarks detection. They obtained accurate and fast results on shapes such as the lungs or the kidneys. Landmark-based methods are generally efficient in both computation time (close to or below 1 s depending on the method and implementation) and accuracy. However one may criticize the difficulty of determining reliable anatomical landmarks. While landmarks may be relatively obvious for bones, finding reproducible landmarks for soft tissues (e.g. the liver) may be more challenging. Instead of detecting landmarks, some researchers proposed to directly localize the ROI containing the organs. These ROI are often represented in the form of bounding boxes. The problem can be solved by classification, which consists in building a classifier and parsing the image to find the best box position. This kind of approach can be performed using the MSL in an independent fashion (Lu et al., 2012; Wu et al., 2014) which may lack spatial consistency. Sofka et al. (2010) enhanced this type of approach by finding the best sequential order of detection thanks to a hierarchical detection network. Alternatively, regression approaches were also proposed. Criminisi et al. (2013); 2011) first proposed to use regression for multiple organs localization. This type of technique was also successfully applied to localization in MR images (Pauly et al., 2011). This method reaches a good trade-off between accuracy and speed (in seconds) and is relatively easy to implement (there exist many random forest open source implementations, see Section 3.2). Following the work of Dollár et al. (2010) on cascades of regression, Cuingnet et al. (2012) proposed to improve the method of Criminisi et al. (2011) with a cascade of regression random forests for kidneys localization. Despite the efficiency and robustness of these methods, one may criticize the rough parameterization of the organs (bounding boxes). We refer the reader to Zhou (2014) for a recent and extensive overview of the different discriminative methods for anatomy detection.

1.2. Background on random forests

A random forest is a collection of decision trees in which randomization is introduced during learning to reduce the statistical dependence between the trees. This concept was first introduced by Amit and Geman (1994); 1997) and Ho (1995); 1998) almost at the same time, and further developed and theorized by Breiman (2001) (with the introduction of bootstrap aggregating). The paternity of decision trees is given to Morgan and Sonquist (1963) who first used regression trees for process explanation and prediction. We refer the reader to Hastie (2009) for details on decision trees and their variants.

The original idea of random forest was to add randomness in the node decision tree construction. Instead of choosing the best split among all possible features, the best split is determined among a subset of randomly chosen features at each node. This considerably decreases the training time, and also has the interesting characteristic of being more robust against over-fitting. The random forest method is particularly adapted to multiple-output regression problems.

Given a set of training samples (called *labels*, which are the values that we want to regress, and their associated features), each tree is learned from a uniformly randomly chosen subset (method called bagging or bootstrap aggregating). Each tree node is built iteratively by finding the best split according to random image feature values and a given objective function (e.g. information gain or residual sum of squares (RSS) of the labels compared to the left and right mean labels). At a new node, if the stopping criterion (e.g. minimum node size) is fulfilled, the tree construction is stopped and a leaf node is built. The training samples that reach that leaf are recorded by storing their mean value or distribution. Each tree thus performs a coherent partition of the initial training subset. During testing, each test sample is pushed through each tree and reaches specific leaves. All these outputs are then combined (e.g. by average or maximum a posteriori) to give the final output prediction. We refer the reader to the book of Criminisi and Shotton (2013) for more details on random forests and their application to computer vision and medical image analysis.

Random forests have practical advantages: they are fast (both in testing and training), they are relatively easy to implement (binary trees), they scale well to large training sets and they give robust and accurate predictors. This explains the recent popularity of this method, in particular in the computer vision community. One competitor for the speed aspect may be the *random ferns* (Ozuysal et al., 2007), a non-hierarchical variant. However it requires large training sets to outperform random forests.

Multiple variants have been proposed to improve the accuracy and/or to integrate more contextual information. In Section 1.1, we already mentioned random forest extensions related to atlas-based frameworks (Konukoglu et al., 2013; Lombaert et al., 2014; Zikic et al., 2013). Hereafter, we give a brief overview of existing variants, with a particular focus on object/organ detection and localization.

Note that those approaches can often be applied with other types of classifiers and regressors than random forests. Separate decision forests may be trained iteratively to refine successively the predictions as in the work of Shotton et al. (2008). Dollár et al. (2010) leveraged this idea of cascaded predictions by learning multiple regressors. Each regressor is learned so as to minimize the difference between the true pose of the object to localize and the pose computed by the previous regressors. Tu (2008) proposed to integrate the semantic information from the first classifier into the next classifier (probabilistic boosting tree classifiers (Tu, 2005)). This approach called *auto-context* was successfully applied to brain segmentation (Tu & Bai, 2010). Semantic context may also be introduced directly in the forest construction. The *entangled forests* of Montillo et al. (2011) are built so that tree nodes may depend on the sub-results of the same tree. Authors show that it improves accuracy and captures more long-range context. In the *context-sensitive forests* proposed by Kontschieder et al. (2012) the forest entanglement is enhanced by sharing both regression and classification information between nodes. Kontschieder et al. (2013) also generalize the idea of entanglement and auto-context with the *geodesic forest* by using long-range soft connectivity features and a new objective function inspired by conditional random fields. Such entangled methods come with a cost: an increasing complexity and a longer learning process (8 h to train the entangled forest on 200 CT volumes for the segmentation of 12 organs (Montillo et al., 2011)), which can make parameter optimization delicate. Joint pixel classification and shape regression have also been successfully applied to multi-organ segmentation (Glocker et al., 2012). Computation speed is however not mentioned. When the objects to localize have a

particular structure (which is the case in medical imaging), the random forest may be conditioned with global variables. Dantone et al. (2012) proposed to tackle the problem of face detection by using the head pose as a prior to learn several random forest regressors. Sun et al. (2012) generalized this idea with a random forest regression model incorporating dependency relationships between output variables through a global latent variable. This has the advantage of improving the accuracy of a classical regression forest at a small computational overhead.

1.3. Conclusion on related works and summary of our contributions

The previous sections showed the richness of existing methods for object localization. However to our knowledge there is no method dedicated to multi-object localization taking into account the object spatial relationships, the contextual information and the shape priors in a fast and accurate manner. Our framework attempts to fill all these requirements. The localization of multiple organs is performed with a cascade of regression forests conditioned with organ-dedicated shape priors. It gives confidence maps for each organ as a localization result. This relates our method to the works of Criminisi et al. (2013; 2011) on random forest regression for anatomy localization, Cuingnet et al. (2012) on cascades of regressors for kidneys localization, and Sun et al. (2012) on conditional regression models. By keeping the method relatively simple to implement, we show that we can get accurate and robust performance at a low computation cost. Our method only requires to learn several classical random forest regressors in cascade, to build one probabilistic atlas per organ and to implement an efficient confidence map computer. We actually propose a fast implementation based on convolution for this last critical component (see Section 2.2.2). The low training computation time also allows us to perform an exhaustive evaluation of the different parameters of our framework (Section 3). While giving us a good understanding of the different parameters contributions, this also guarantees the best possible accuracy. We show in Section 4 that our method provides in a few seconds a consistent and accurate localization for multiple organs in 3D medical images.

2. A method for localizing multiple organs

The method we present here is designed for the localization of multiple organs. In Fig. 1 we present an outline of the method for the localization of the two kidneys. The method works in two steps, in a global-to-local fashion. It makes use of three main components: (i) bounding boxes predictors (in our case: regression random forests), (ii) probabilistic atlases for each organ, (iii) confidence maps. Confidence maps combine votes aggregation with a probabilistic atlas, resulting in organ-dedicated probabilistic maps. The first step of our method aims at finding the global organ locations with a single regressor while the second step aims at refining individually the organ positions with organ dedicated regressors. After introducing our main notations in Section 2.1, we explain the principle of confidence maps in Section 2.2 and we detail the cascade approach in Section 2.3.

2.1. Notations

We use the following notations hereafter:

- $\Omega \subset \mathbb{R}^3$ is the image domain,
- $o \in [\![1, N_{\text{org}}]\!]$ are the indices of the N_{org} organs to localize,
- $k \in [[1, K]]$ is the index for one single prediction/regression,
- $G = S \circ T$ is a geometric transformation of $\mathbb{R}^3$ composed of a translation T and an anisotropic scaling S. $G_{o,k}$ denotes the geometric transformation associated with organ o and regression k,
- $\mathbf{b} \in \mathbb{R}^6$ denotes a 3D bounding box parameterization either as its center location and size ($\mathbf{c}$, $\mathbf{s} \in \mathbb{R}^3$) or as its extremal vertices ($\mathbf{b}_{\min}, \mathbf{b}_{\max} \in \mathbb{R}^3$), $\mathbf{b}_{o,k}$ then denotes the bounding box parameterization of organ o and regression k,
- $\mathbf{x} \in \Omega$ denotes any voxel of an image I,
- $\mathbf{v_k}$ denotes a test voxel for regression k.

2.2. Merging shape priors and vote distributions with confidence maps

In the main works on organ localization with regression, spatial vote distributions are not fully exploited. However, as shown with the Hough forests (Gall & Lempitsky, 2009), vote aggregation can give more information than a single measure of the distribution (e.g. mean or median). For this purpose we introduce the notion of *confidence map*, which encodes the confidence in finding a target organ at a given location. It is built through an aggregation process, making use of both the spatial distribution of the regression votes and of organ shape priors through probabilistic atlases.

2.2.1. Probabilistic atlas building

To compute a probabilistic atlas of an organ we first register the binary masks of several samples of this organ. For this purpose let $\{M_i\}_{i \in [\![1, N_{\text{m}}]\!]}$ be the set of N_{m} different cropped binary masks of the organ such that $\forall i, 1 \leq i \leq N_m$ M_i; $\Omega \rightarrow [0, 1]$ and where M_r ($r \in [[1, N_{\text{m}}]]$) is a mask of reference chosen arbitrarily. We transform each mask with a transformation Ψ_i (rigid and anisotropic scaling) in order to scale up the masks M_i to the same size as M_r (considering the bounding boxes of the binary shapes). The probabilistic atlas A is then computed as an average of these masks:

$$A(\mathbf{x}) = \frac{1}{N_{\text{m}}} \sum_{i=1}^{N_{\text{m}}} M_i \circ \Psi_i(\mathbf{x})$$

Each value $A(\mathbf{x})$ evaluates the probability of a voxel $\mathbf{x}$ to belong to the organ. Fig. 2a and d show probabilistic atlases for the liver and the right kidney respectively. We normalize the atlas in location and scale in the range $[-1, 1]^3$. Thus if $\mathbf{x} \notin [-1, 1]^3$ we have $A(\mathbf{x}) = 0$.

2.2.2. Confidence map principle and numerical considerations

Here we explain the construction of the confidence maps and propose an implementation to compute them efficiently. The principle is illustrated for the kidney localization in Fig. 3. Numerous voxels of the image vote for the kidney bounding box location and each of these votes are combined with the right kidney probabilistic atlas (following an aggregation process).

Organs bounding boxes prediction. Let us consider a regressor $\boldsymbol{R}$ predicting the bounding box position for organ o. For a test voxel $\mathbf{v}_k$ and its associated image features from an image I, it gives a prediction of the bounding box $\mathbf{b}_{o,k}$ of an organ o. Note that, as in Criminisi et al. (2013), the label regressed is actually the relative distance of the voting voxel to the extremal vertices of the bounding box. For the sake of comprehension, we simplify the notation and assume that the regressor predicts the bounding box parameters. If the regressor is a random forest one can store the distributions of the training labels in the leaves of the trees. During the testing phase this allows us to compute the posterior probability $p(\mathbf{b}_{o,k}|I, \mathbf{v}_k)$ that we denote $\alpha_{o,k}$. It gives a confidence score about the vote of $\mathbf{v}_k$ for the location of the bounding box $\mathbf{b}_{o,k}$ of the organ o. We can thus write $[\mathbf{b}_{o,k}, \alpha_{o,k}] = \boldsymbol{R}(\mathbf{v}_k, I)$.

Confidence map construction. If we consider the localization at a voxel level in the image, either the voxel $\mathbf{x}$ belongs to the box or it does not. The probability $p(\mathbf{x} \in o|\mathbf{b}_{o,k})$ of a voxel to belong to the organ o given the predicted bounding box $\mathbf{b}_{o,k}$ then follows a uniform distribution. This approximation is rough, especially for organs with non convex shapes as the liver for example. Thus we propose to enhance this hypothesis by using a shape prior through the use of probabilistic atlases A_o of organ o. We assume that $p(\mathbf{x} \in o|\mathbf{b}_{o,k}) = A_o \circ G_{o,k}(\mathbf{x})$ where $G_{o,k}$ is the transformation which translates and resizes (anisotropically) the bounding box $\mathbf{b}_{o,k}$ to fit into $[-1, 1]^3$.

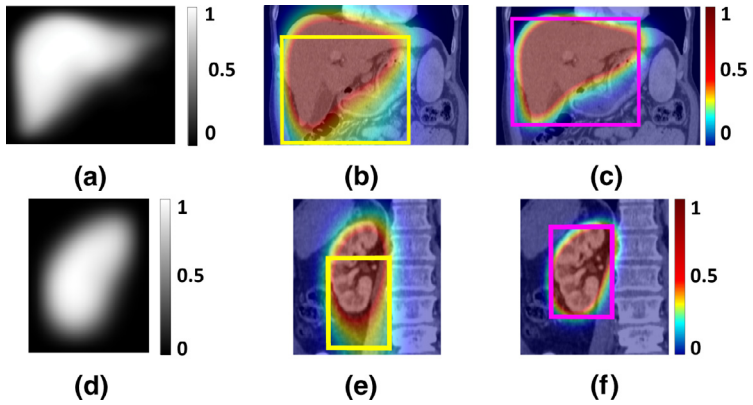

Fig. 2. Atlases of the liver (a) and the kidney (d), localization of the liver and the right kidney: images with confidence map as overlay (images have been cropped) and predicted median box after global step (b), (e) and local step (c), (f).

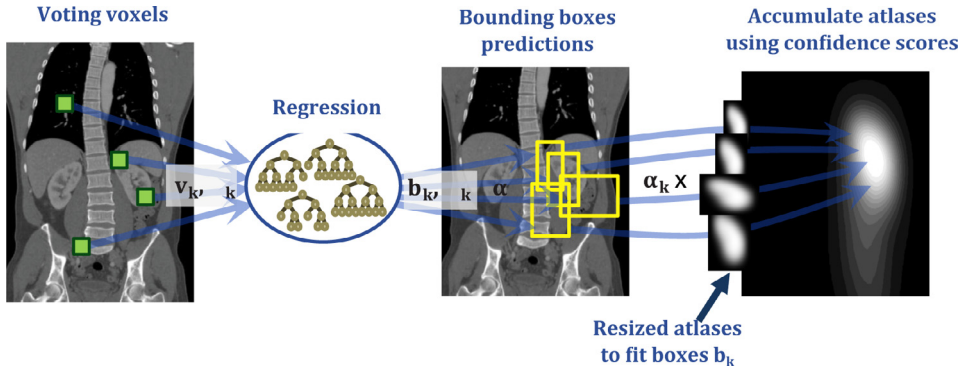

Fig. 3. Outline of the confidence map computation, here illustrated on an application for the only right kidney localization (for the sake of understanding).

We want to know the joint probability of localizing an organ and that a voxel of the image belongs to this organ, which means that we want to compute the probability of having both a predicted bounding box $\mathbf{b}_{o,k}$ and $\mathbf{x} \in o$, given an image I and a test voxel $\mathbf{v}_k$:

$$p(\mathbf{x} \in o, \mathbf{b}_{o,k}|I, \mathbf{v}_k) = p(\mathbf{x} \in o|\mathbf{b}_{o,k}, I, \mathbf{v}_k).p(\mathbf{b}_{o,k}|I, \mathbf{v}_k)$$

As to deduce the probability of $(\mathbf{x} \in o)$, knowing the bounding box $\mathbf{b}_{o,k}$ or knowing the image I and the voting voxel v_k are equivalent $(p(\mathbf{x} \in o|\mathbf{b}_{o,k}, I, \mathbf{v}_k) = p(\mathbf{x} \in o|\mathbf{b}_{o,k}))$, we can consider that the random variables $(\mathbf{x} \in o|\mathbf{b}_{o,k})$ and $(I, \mathbf{v}_k|\mathbf{b}_{o,k})$ are independent conditionally to $\mathbf{b}_{o,k}$. We can write:

$$\begin{aligned} p(\mathbf{x} \in o, \mathbf{b}_{o,k}|I, \mathbf{v}_k) &= p(\mathbf{x} \in o|\mathbf{b}_{o,k}).p(\mathbf{b}_{o,k}|I, \mathbf{v}_k) \\ &= \alpha_{o,k} \,.\, A_o \circ G_{o,k}(\mathbf{x}) \end{aligned}$$

In the work of Criminisi et al. (2013) and Cuingnet et al. (2012), the authors assume that the localization results from a measure over the predictions given by multiple test voxels of the image (maximum a posteriori, average or median). Here we consider the entire set of K predictions and model the final confidence score F in each voxel of the image I, defining the confidence maps of organ o as follows:

$$F(\mathbf{x} \in o, \{\mathbf{b}_{o,k}\}_{k \in [\![1,K]\!]}|I) = C_o(\mathbf{x}) = \frac{1}{K}\sum_{k=1}^{K} \alpha_{o,k}.A_o \circ G_{o,k}(\mathbf{x})$$

where C_o is the confidence map of organ o. We then normalize the map C_o by its maximum to get values in the range [0, 1]. For a voxel $\mathbf{x}$ of the image I, $C_o(\mathbf{x})$ can be seen as the chance of $\mathbf{x}$ to belong to the organ o.

Some examples of confidence maps are given in Fig. 2(b), (c), (e) and (f). Fig. 2(b) and (e) show that the maps capture the ambiguity of the vote distribution, as we observe that some voxels were correctly voting for the box upper wall position, an information that the median was not able to capture.

Fast confidence map implementation. The confidence map C for a given organ is actually built by translating and scaling the organ probabilistic atlas A according to each vote k and accumulating the result in C with weight α_k. The map C gives a confidence score about the presence of the organ at a given location in the image.

The computation of these maps may be expensive. Therefore we propose a fast implementation which considerably reduces the computation time while not degrading significantly the accuracy (the pseudo-code is given in the Appendix in Algorithm 1). The idea is to uniformly discretize the space of predicted bounding boxes dimensions (over each dimension x, y and z). With K votes, each corresponding to a bounding box location and size $\mathbf{d}_k$, we can compute the corresponding centers $\mathbf{c}_k$. We denote $\mathbf{d}_{\min}$ and $\mathbf{d}_{\max}$ the minimum and maximum box sizes over the K predictions. Assume we want to discretize the space of these box sizes ensemble. We do it by defining the jth (where $j \in [[1, J]]$) discretized size $\hat{\mathbf{d}}_j \in \mathbb{R}^3$ as:

$$\hat{\mathbf{d}}_j = \frac{j}{J}(\mathbf{d}_{\max} - \mathbf{d}_{\min})$$

Each size dimension is discretized the same way. For each possible discretized size of box $\hat{\mathbf{d}}_j$, we create a volume C_j of the same size as the image (it is however possible to work at a lower resolution) initialized to 0. For each prediction we set the box centers to their confidence score value in the volume C_j. We define the probabilistic atlas A_j which has been anisotropically rescaled to the jth sample size. This atlas is then convolved with the volume C_j. The final map C is computed as the sum of each C_j and normalized by the maximum to get the final probability map. With a uniform discretization on 27 box sizes (three per spatial dimension, which is good trade-off between speed and precision), the computation is about 30 times faster for a limited loss of accuracy (see Section 4). These two possible implementations are

also detailed and compared in our previous publication (Gauriau et al., 2014).

2.3. Localizing organs with cascade of regressors and confidence maps

Our approach consists of two steps in a coarse-to-fine fashion. A first regressor, aiming at capturing the spatial relationships between the organs, is learned using global information: voxels of the whole image vote for the positions of all the organs simultaneously (all the bounding boxes are regressed with the same forest, as done by Criminisi et al. (2013)). Afterward, new regressors, dedicated to each organ, are learned using more local information. We may wonder why not to localize directly each organ independently like in the second step. Our experiments showed that the global step helps increasing the robustness of the approach as it encodes the global anatomy prior. The benefit of the cascade approach has also already been shown in Cuingnet et al. (2012). The authors localize the two kidneys by using a cascade of random forest regressions. The first regression predicts the two bounding boxes simultaneously. In a second step, two regressors refine the left and right kidneys bounding box centers independently. For these two last regressors, the voting voxels (at training and testing time) are selected in a ROI around the bounding box centers predicted by the first regressor.

Here we propose to introduce the use of confidence maps for refining the votes in the cascade of regressors. Moreover, instead of only refining the bounding box centers in the second step, we refine the prediction of all the bounding box parameters. The selection of voxels which vote in the local step may benefit of the information given by the confidence maps, that is to say the vote distributions and the shape prior. Note that this method could be applied with other types of multi-variate regressor. In Section 3 we detail how to implement this method with the random forest as a regressor.

2.3.1. Global step

In the first step, a random subset of K_g voxels $\{\mathbf{v}_k\}_{k\in[\![1,K_g]\!]}$ of the image I vote for the bounding boxes parameters $\{\mathbf{b}_{k,o}\}_{k\in[\![1,K_g]\!]}$ of all organs o. By regressing all the parameters jointly (i.e. $[\{\mathbf{b}_{o,k}\}_o, \{\alpha_{o,k}\}_o] = \boldsymbol{R}(\mathbf{v}_k, I)$), the relationships between the organs are implicitly embedded during learning. Votes are performed according to long-range features computed from the image (see Section 3 for the details on the features). These features are chosen to encapsulate global information from the image. Then the confidence map C_o for each organ o is computed using the algorithm described in Section 2.2.2, given the probabilistic atlas A_o. Fig. 2c and f shows some examples of results after this global step for the liver and the right kidney localization. In these figures we compare the localization with the confidence maps and the one given by the bounding boxes (computed from the median over the predictions). We see that capturing the distribution of the votes can be more informative than just a median.

2.3.2. Local step

The second step aims at improving the previous localization. Each organ o is re-localized individually using a ROI selected thanks to the previous global localization and the resulting confidence maps. The ROI is found by computing the binary mask B_o from the map C_o thresholded at a value t_g (see Section 3 for numerical values). Afterward we select a random subset of K_l voxels $\{\mathbf{v}_k\}_{k\in[\![1,K_l]\!]}$ such that each voxel is in the ROI ($B_o(\mathbf{v}_k) = 1$). Each voxel $\mathbf{v}_k$ votes for the bounding box of organ o using a regressor specifically trained for this organ. Contrarily to the previous step this regressor is now learned using shorter-range features (see Section 3) and computed in the vicinity of the organ o thanks to the confidence map. This gives more importance to local information. Then the votes are used to compute new and more accurate confidence maps C'_o for each organ. Fig. 2c and f shows the benefit of adding this local step after the global step: the confidence maps are more focused on each organ and less spread around it.

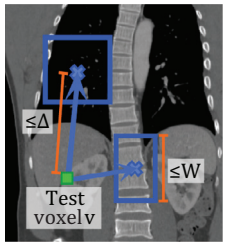

Fig. 4. Example of 1D features computed at each tree node: difference of mean intensities in two patches of random size and location in a certain range.

3. Using the random forest as a regressor: implementation and off-line training

To test our approach we propose to localize six abdominal organs: the liver, the right and left kidneys, the gallbladder, the spleen and the stomach from various types of 3D CT volumes and using the random forest as a regressor. When using the random forest there are multiple parameters to set. In this part we show that a simple parameters optimization procedure helps to get better performances. These experiments also demonstrate the good robustness of the algorithm to small variations of these optimal parameters.

3.1. Using the regression forest as a regressor

To regress the parameters $\mathbf{b}_o$ (see notations in Section 2.1) we choose to use the random forest as in Criminisi et al. (2013). The authors showed that the regression random forest is well suited to rough localization especially in terms of accuracy and speed. We are able to reinforce its robustness and accuracy using our global-to-local approach with confidence maps.

We use the same kind of features as Criminisi et al. (2013), similar to those introduced by Gall and Lempitsky (2009) and Shotton et al. (2009). These contextual features are defined as the difference of mean intensities in two 3D patches of random size and random location in a given range. We define Δ as the maximum offset in each dimension of the patches to the test voxel and W as the maximum width size of the 3D patches. Fig. 4 shows an example of this type of features. Note that the patches may be reduced to a single voxel or an empty box, thus giving unary features. For the global step of the algorithm, long-range features (high values of Δ and W) are used (they are computed from the image after Gaussian smoothing, with kernel width of 3 mm). For the local step, more local features (shorter values of Δ and W) are used. In each tree leaf the multivariate Gaussian distributions of the parameters to regress are stored. This statistical information is used to compute the confidence scores of each vote.

The first global forest is learned from a random subset of voxels from images of the training set. After learning this forest we proceed to the localization on the same training images and we compute the confidence maps associated with this result. Note that in practice the votes with low confidence scores are discarded. To compute the confidence maps we only keep the best votes according to the confidence score (if N is the number of votes, $K \leq N$ is the number of votes that we keep). Then each local regression forest is learned from training voxels in the vicinity of the corresponding organ thanks to the confidence maps (hypothesis demonstrated in next experiments). After the forest of the global step is trained, we use it to test the training images. The resulting confidence maps are then thresholded to select the training voxels that are used to train the local forests. The testing phase proceeds exactly the same way.

Table 1
Regression forest parameters.

Parameters	Maximum tree depth	Nb training voxels/image	Bootstrap size	Range of features (mm)	Confidence map threshold	Nb of votes	Nb of best votes
Global step	$T_g = 14$	$S_g = 25,000$	$B_g = 40\%$	$\Delta_g = 50, W_g = 180$		$N_g = 60,000$	$K_g = 10\%$
Local step	$T_l = 14$	$S_l = 10,000$	$B_l = 50\%$	$\Delta_l = 40, W_l = 40$	$t_{\text{cmap}} = 40\%$	$N_l = 10,000$	$K_l = 10\%$

3.2. Database description and implementation

The database we use is the same as in our previous work (Gauriau et al., 2014). It is composed of 130 3D CT images coming from 112 patients with diverse medical conditions (healthy and pathological subjects, no organ missing). It includes volumes with varied fields of view, body shapes, resolution and use or not of contrast agents. Slices and inter-slices resolutions range from 0.5 to 1 mm and from 0.5 to 3 mm, respectively. For both training and testing we work at a resolution of 3 mm. All the organs have been manually segmented in these 130 volumes. The dataset is split randomly into 50 and 80 volumes for training and testing, respectively. Our method is implemented in C++ and running times are given for a machine with two 2.3 GHz cores and 8 Go RAM.[1] The forest training is parallelized (one tree per processor).

3.3. Off-line training procedure

To reach the best performance and analyze each aspect of the algorithm, we perform an extensive optimization of the algorithm parameters. Considering the computation time (learning one tree of depth 12 takes about 2 min), we choose to use a greedy and sequential parameters optimization.

Protocol. For each parameter we perform a five-fold cross-validation on the training set (40 for training and 10 for testing). For computation time and memory reasons, the accuracy of the algorithm is measured as the mean over each organ of the mean distances of the predicted boxes to the ground truth bounding boxes. We assume this would give nearly equivalent results if we looked at the accuracy of the confidence maps (as the goal is to reduce the bias and variance of the random forest regressors). Moreover, our experiments showed that optimizing the parameters per organ would not bring significant improvements. The predicted boxes are simply computed by taking the median of the votes over the best votes K_g and K_l. For each test image we run the algorithm five times to be able to measure and reduce the variability induced by the randomness of the method. For learning the forests we perform subbagging (Andonova et al., 2002) with uniform random draw with replacement. This means that we learn each tree on a random subset of the training set the size of which is inferior to the training set size (in the original bagging method, the bootstrap has the same size as the whole training set). If subbagging is interesting to reduce the training computation time, we will also show that it can also make the algorithm more robust. The node feature selection is also randomized (we compute only 30 random features at each node). For each experiment we use seven trees, as this gives a good trade-off between low variance and computation time (also shown in later experiments, see Fig. 12). Before each learning we de-correlate the data with a whitening transform. This is a traditional statistical method which forces the variance of the training data along each dimension to become equal to one, thus preventing to give more importance to one or another dimension during learning.

[1] Due to industrial confidentiality concerns we are not able to release our own code. Several packages and libraries on random forests can be found on the web such like the scikit-learn package in Python (Pedregosa et al., 2011) or TMDA in C++ (Hoecker et al., 2007). GPU implementations have also been proposed (Grahn et al., 2011; Sharp, 2008).

We first initialize every parameter arbitrarily with values already giving reasonable results. These values were quite easy to find, taking into account the literature on the subject and a few experiments. Then we optimize each parameter one-by-one by grid-search and we replace its value by the optimized one.

Parameters to be optimized. We use the index g for the global forest and l for the local forests.

For the global forest we study the following parameters (in this order):

- T_g, S_g: the maximum tree depth and the number of voxels per image used for training respectively (studied jointly),
- B_g: the bootstrap size (percentage of training samples used to learn one tree),
- Δ_g, W_g: feature patches maximum offset and width (studied jointly),
- N_g, K_g: the number of test voxels per image and the percentage of test voxels kept for the final prediction computation respectively (studied jointly).

For the local forests we study the same parameters: T_l, S_l, Q_l, Δ_l, W_l, N_l and K_l.

We also study the parameter t_{cmap} which is the threshold of the confidence map computed after the global step for selecting the voxels used for learning or testing the local forest.

3.4. Off-line training results

All the final results of the parameters optimization are given in Table 1.

The graphics of Figs. 5 and 6 show the optimization of the parameters of the global step. Each of them gives the testing cross validation accuracy computed for the different parameter values in the same order as they were optimized. Fig. 5a shows that the accuracy is increasing with the maximum tree depth until $T_g = 14$, and for this value the optimal number of training voxels per image is $S_g = 25,000$. The maximum tree depth and the number of training voxels per image being fixed, we can look at the optimal bootstrap value. In Fig. 5b we see that an optimal value is around $B_g = 40\%$ although close values give similar accuracy. We note that a bootstrap of 100% (equivalent to original bagging) gives the same result but with higher variance (mean variance over all the organs). Here the subbagging helps to achieve more robustness.

In Fig. 5c we can see the influence of the feature parameters. It shows the algorithm accuracy in function of the maximum offset value Δ_g of the 3D patches. Each curve is given for different values of the patches maximum width W_g. We see that big patches lead to better accuracy, reaching a plateau at about $W_g = 60$ mm of maximum width. We also note that increasing values of Δ_g degrades the accuracy. Some experiments show that this can be explained by an increasing number of patches that fall outside of the image domain, the features being then undetermined. In our implementation we do not discard labels with undefined features. At a node, labels with undefined features just go right and left in the tree, as these labels may have defined features at other nodes of the tree and then be useful. While former results were related to the training part, the next figure shows results on the testing part. For this test we keep the best forest given after the last parameter tuning. In Fig. 6 we show the accuracy compared

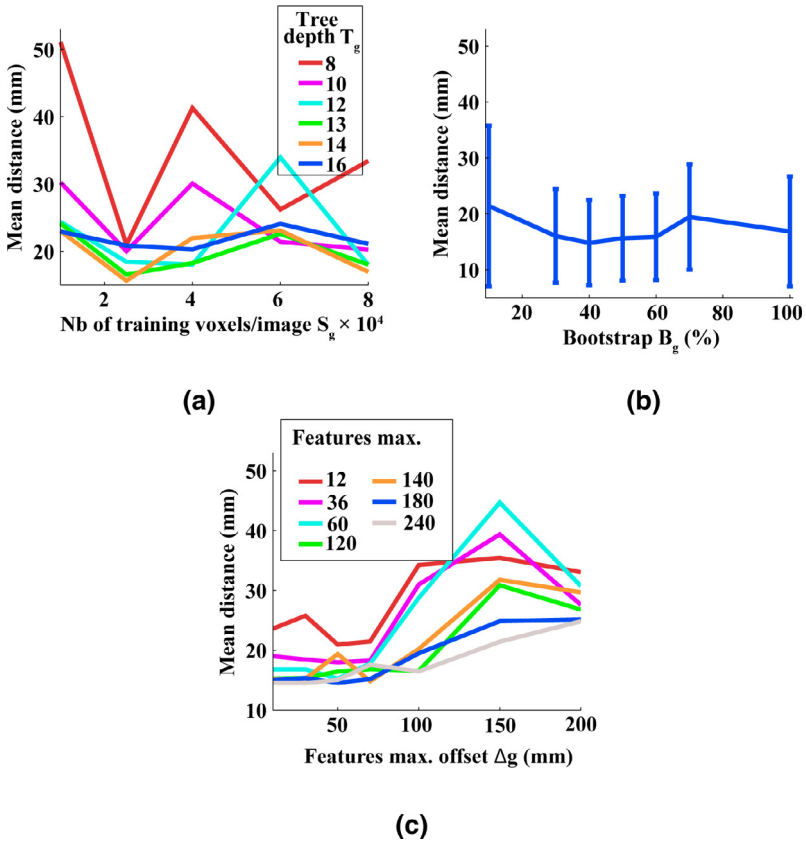

Fig. 5. Parameter optimization results for the global random forest. Performance is given for (a) the number of training voxels per image S_g, each curve corresponding to a given maximum tree depth, (b) the bootstrap B_g with error bars corresponding to the standard deviation of the localization accuracy, (c) the features parameter Δ_g and W_g.

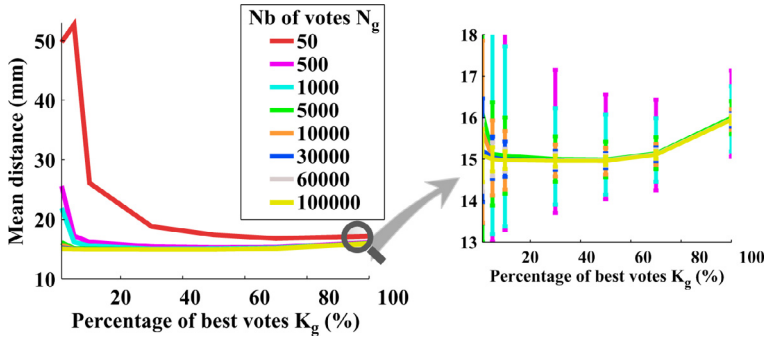

Fig. 6. Performance in function of the percentage of selected test voxels K_g, each curve corresponding to different values of the number of test voxels N_g. The right plot is a zoom with error bars related to the standard deviation of the results over the multiple tests.

to the percentage of best test voxels K_g. Each curve corresponds to different numbers of test voxels per image N_g. We note that from $N_g = 1000$ we reach almost the same accuracy. However, looking at the right plot which is a zoomed version showing the standard deviation over the multiple tests, we see that the standard deviation over the multiple tests decreases when the number of test voxels increases. With $N_g = 60,000$ and $K_g = 10\%$ the results reach a standard deviation below 0.2 mm. We deemed such performance reasonable enough, as increasing this value would also increase the computation time.

The graphics of Figs. 7 and 8 show the optimization of parameters for the local random forests (second step in the cascade). In particular Fig. 7a shows that from a maximum tree depth of $T_l = 14$ the convergence is reached, and that a number of $S_l = 10,000$ training voxels per image gives optimal results. Fig. 7b shows a very good robustness with respect to the bootstrap value, and we set it to $B_l = 50\%$. Fig. 7c shows the accuracy in function of the feature parameters. Note that more local features than in the global forest are preferred. This confirms the interest of using a coarse-to-fine approach. As highlighted in Zhou et al. (2010), such a framework mimics the human visual system in a certain way: global information is first used to define the context and then, while restricting the ROI, more local information is then required to refine the understanding analysis of this area.

For the best performance, these features should have displacements up to $\Delta_l = 40$ mm and patches widths up to $W_l = 40$ mm. Finally in Fig. 8 we show the results in function of the testing voxels for the local part. From $N_l = 1000$ we get similar accuracies, but looking at the right plot we set $N_l = 10,000$ and $K_l = 10\%$, which corresponds to a standard deviation in test below 0.2 mm. Finally, Fig. 7d shows the performance result in function of the confidence map threshold t_{cmap}. This parameter actually controls the size of the ROI in which the local step is performed. Setting the value to

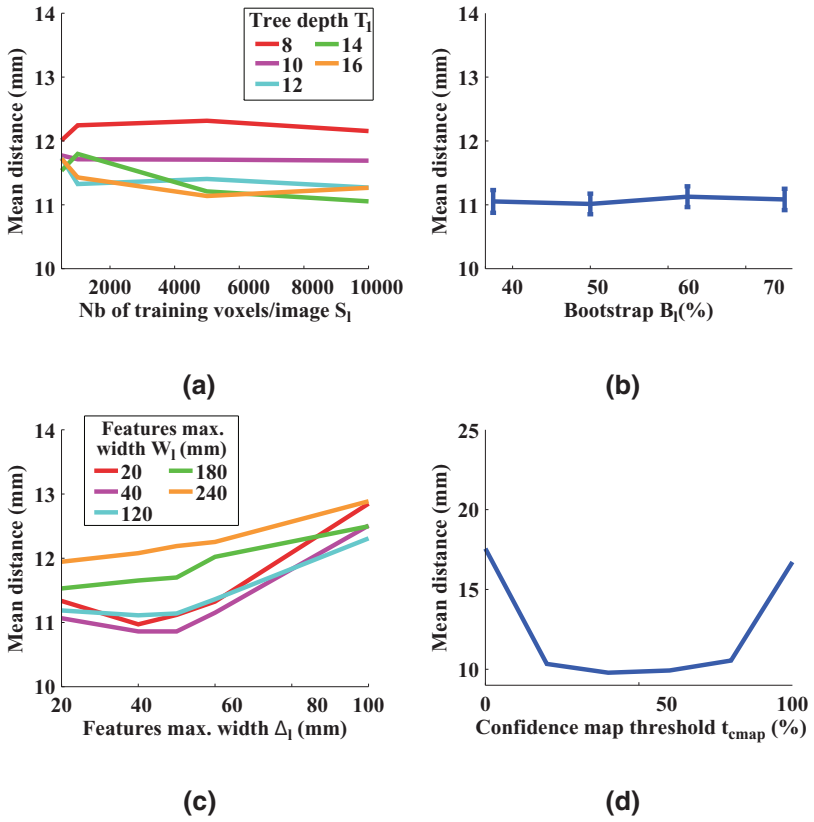

Fig. 7. Parameter optimization results for the local random forests. Performance is given for (a) the number of training voxels per image S_l, each curve corresponding to a given maximum tree depth, (b) the bootstrap B_l with error bars corresponding to the standard deviation over the localization accuracy, (c) the features parameter Δ_l and W_l, (d) the threshold of the confidence map for local forest voxel selection.

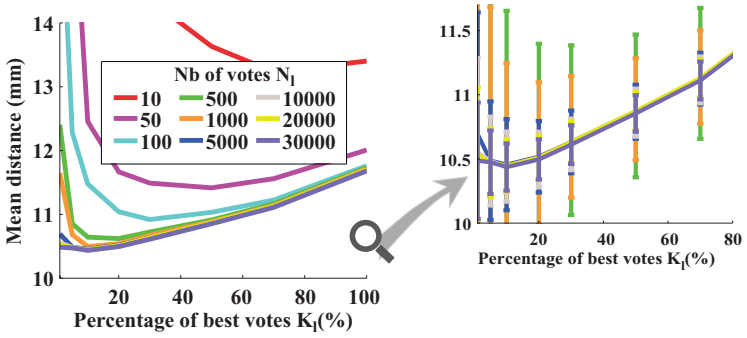

Fig. 8. Accuracy in function of the percentage of selected test voxels K_l, each curve corresponding to different values of the number of test voxels N_l. The right plot is a zoom with error bars related to the standard deviation of the results over the multiple tests.

$t_{cmap} = 40\%$ gives the best results, but we note that this parameter is very stable and values between 20% and 80% give similar results (this is also explained by the relatively small spread of the confidence maps, see Fig. 2c and f. This result demonstrates the interest of restraining the ROI in the cascade approach.

3.5. Conclusions on parameters optimization

Although very costly in terms of computation time (more than 2 full weeks in cumulative time), this study on parameters is very informative. First of all, it shows that the choice of parameters may be crucial for having a good algorithm accuracy. Fortunately, our results also show that there is no need to tune every parameter very precisely, since the results remain quite stable in certain range of parameter values. This proves that our implementation is robust. This will be also demonstrated by the results presented next, showing that the same parameter optimization can be used for a varied and large set of images. The experiments also showed the interest of using subbagging. While reducing the training computation time, it also reinforces the robustness of the algorithm. Finally this part also confirms the intuition that the range of features directly relates to the level of information we want to retrieve. The coarse global step localization requires features covering large parts of the image, as if we had to look at the image from far away. On the other hand, more local features are needed in the local step for refinement purpose: the ROI is narrower and we look at the image more closely.

4. Evaluation and validation of the approach

In this section we evaluate our approach, taking into account the parameters optimization realized in the previous section. We perform here two types of experiments. The first one aims to show that the

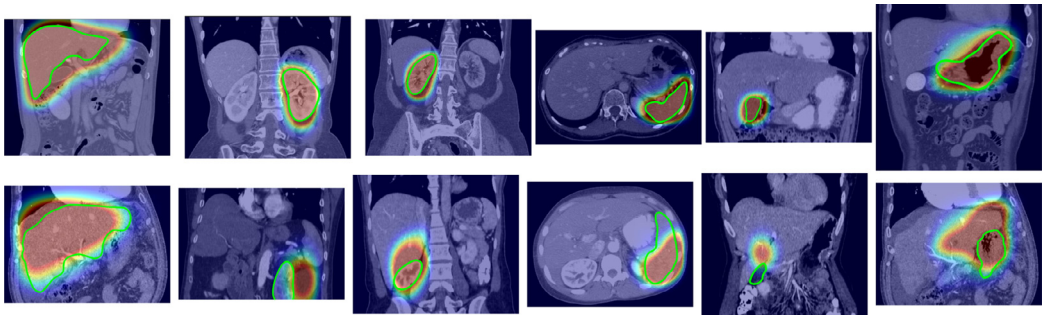

Fig. 9. Best and worst results per organ (first and second line respectively) according to the mean distance to ground truth. Confidence maps are overlaid on the images and the green contour corresponds to the manual segmentations. From left to right, organs are in this order: liver, left and right kidneys, spleen, gallbladder and stomach. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

confidence maps can be used as a result in itself, while the second experiment demonstrates several aspects of our work: the interest of the cascade approach with a comparison to other state-of-the-art methods, the benefit of a parameter optimization, and finally the interest of using the confidence maps in the cascade for the voting voxels selection.

4.1. Protocol of the experiments

For the two following sets of experiments, we use the same database as the one described in Section 3.2. We learn the forests on the training set, following the conclusions of the parameters optimization of Section 3, and we perform our experiments on the testing set (80 volumes). For the first experiment we consider the confidence maps as a result, which corresponds to our own original approach. In the second experiment we want to see the influence of the use of the confidence maps in the cascade and compare the obtained results with those of existing methods. Therefore we use the median bounding boxes as a final result as in Cuingnet et al. (2012).

4.2. Processing time

Implementation details are given in Section 3.2. Training the seven forests (one for the global step and six for the local step) and building the probabilistic atlases takes approximately 20 min. Testing a classical abdominal image (approximate size: 512 voxels × 512 voxels × 230 voxels) takes around 5 s. The confidence maps computation takes about half that time (one map computation takes around 200 ms).

4.3. First experiment

Our first objective is to show that the confidence maps can be used as a localization result itself, giving more consistent information than the bounding boxes alone. Confidence maps are computed at a 5 mm isotropic spacing. In Fig. 9 the best and the worst results are shown for each organ according to the mean distance to ground truth. Moreover an exhaustive visualization of the results can be seen in the **Supplementary material**.[2]

The results of the first experiment are given in Table 2 using the fast implementation of confidence maps, as the results of Gauriau et al. (2014) already shown equivalent performances with the naive and slower implementation. On the first line the results are given of the mean distance (in millimeters) of the thresholded confidence maps contours (with threshold set to 50%) to the ground truth contours. The results statistics (median and standard deviation) confirm that our method is robust to the variety of test images. Fig. 10 details these results by giving the distribution of mean distances to ground truth

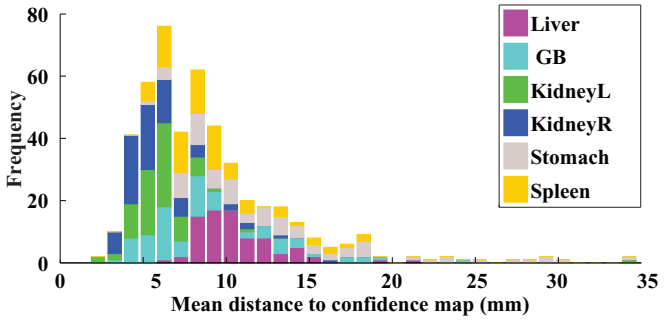

Fig. 10. Histogram of results on confidence maps (mean distance to thresholded maps to 50%), frequencies are stacked over each other.

for each organ. Note the low spread of the results and the very few number of outliers (no result below 35 mm). These results confirm that our approach can be very useful in various contexts such as segmentation initialization or fast anatomy detection, added to the fact that it runs in about 5 s. This makes our approach adapted to clinical applications, especially as the code may still be optimized.

The goal of this first experiment is also to show that the confidence maps give much more information than a simple binary mask or contour. For that purpose we propose adapted evaluation measures taking into account the fuzziness of the maps. If C denotes a confidence map and B a binary mask of the organ ground truth, then the true positive values TP, the false negative values FN and the false positive values FP are defined as:

$$TP = \sum_{x\in\Omega} B(x)C(x)$$

$$FN = \sum_{x\in\Omega} B(x)(1 - C(x))$$

$$FP = \sum_{x\in\Omega} (1 - B(x))C(x)$$

Then following the definitions of the sensitivity (or recall) S, the precision P and the dice coefficient D:

$$S = \frac{TP}{TP + FN},\ P = \frac{TP}{TP + FN},\ D = \frac{2TP}{2TP + FP + FN}$$

we are able to propose weighted versions of these measures. The corresponding figures are reported in Table 2. In Fig. 11 we also show the receiver operating characteristic (ROC) curves for varying confidence map thresholds. We note a good compromise between sensitivity and precision. In terms of accuracy, results are deemed as satisfying for localization purposes. The accuracy is lower for the stomach and the gallbladder which are challenging organs due to their shape and location variability. However the results given in Table 2, Figs. 11 and 10

[2] http://perso.telecom-paristech.fr/~gauriau/MOLoc_SupMat.html

Table 2
Results with confidence maps (5 mm isotropic spacing) with mean distance (mm), dice (%), weighted dice (%), weighted sensitivity (%) and weighted precision (%) (mean ± standard deviation (median)). Mean distance and weighted dice are computed given the confidence map thresholded to 50%.

	Liver	L. Kidney	R. Kidney	Spleen	Gallbladder	Stomach	All organs	Time
Mean distance (mm)	10.6 ± 2(10)	6.3 ± 4(6)	5.8 ± 2(5)	9.1 ± 3(8)	8.8 ± 4(8)	13.9 ± 7(12)	9.0 ± 2(8)	
Dice (%)	74.9 ± 5(76)	69.6 ± 11(72)	70.8 ± 9(73)	60.3 ± 12(62)	33.1 ± 18(34)	48.3 ± 14(50)	59.5 ± 7(62)	
W. dice (%)	74.9 ± 10(79)	75.5 ± 8(78)	77.4 ± 6(79)	72.7 ± 8(74)	51.7 ± 16(56)	58.1 ± 10(60)	68.4 ± 14(73)	5 s
W. sensitivity (%)	78.3 ± 6(80)	69.5 ± 8(71)	70.9 ± 8(72)	69.3 ± 11(71)	51.7 ± 17(54)	52.7 ± 10(54)	65.4 ± 14(69)	
W. precision (%)	67.2 ± 19.8(77)	80.3 ± 11(83)	83.2 ± 8(87)	74.6 ± 14(76)	54.6 ± 18(58)	62.0 ± 17(62)	70.3 ± 18(75)	

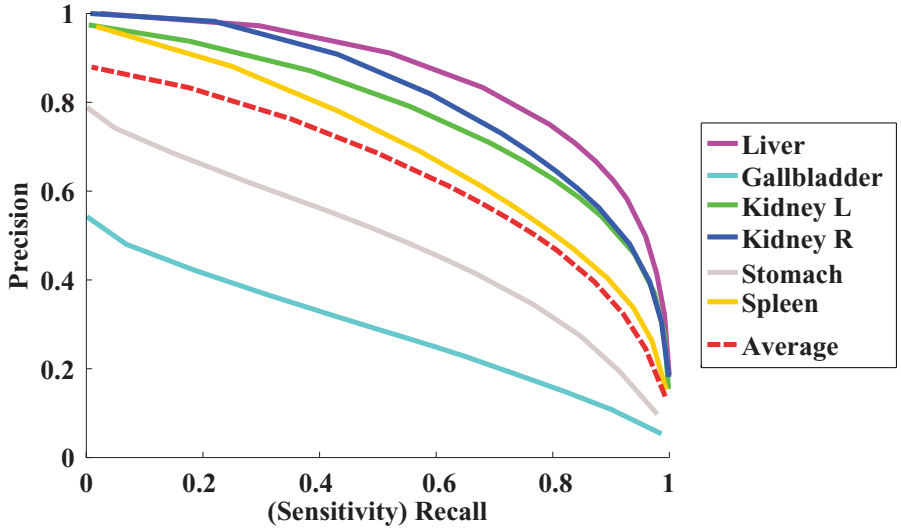

Fig. 11. Receiver operating characteristic (ROC) curves showing how the precision and sensitivity change depending on the confidence map threshold.

Table 3
Box walls mean distances per organ (mean distance (mm) ± standard deviation (median)), per method and per experiment. In bold are the best results. The results are given for the same database (see Section 3.2).

Method	Liver	L. Kidney	R. Kidney	Spleen	Gallbladder	Stomach	All organs	Time(∼) (s)
Criminisi et al. (2013)	14.0 ± 5(14)	12.3 ± 7(11)	13.2 ± 6(12)	14.2 ± 6(13)	15.5 ± 8(14)	16.8 ± 16(16)	14.4 ± 7(13)	0.5
Cuingnet et al. (2012)	12.2 ± 4(12)	6.8 ± 6(6)	6.4 ± 4(5)	9.0 ± 5(8)	11.4 ± 8(10)	14.2 ± 7(13)	10.0 ± 7(9)	2
Our method								
After global step	13.5 ± 4(13)	11.3 ± 6(10)	12.8 ± 5(12)	13.0 ± 5(13)	15.1 ± 6(14)	15.1 ± 6(14)	13.5 ± 6(13)	0.5
After local step, method 1	11.7 ± 4(11)	5.8 ± 4(5)	6.2 ± 3(5)	8.7 ± 4(8)	9.5 ± 4(9)	13.4 ± 6(13)	9.2 ± 5(8)	2
After local step, method 2	**10.7 ± 4(10)**	**5.5 ± 4(5)**	**5.6 ± 3(5)**	**7.9 ± 4(7)**	**9.5 ± 4(8)**	**13.2 ± 5(13)**	**8.8 ± 5(8)**	3.2

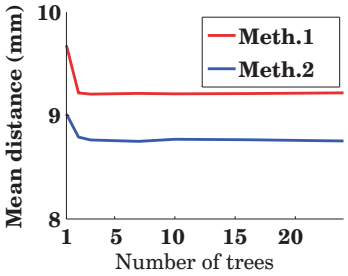

Fig. 12. Results on bounding boxes for methods 1 and 2 and for various numbers of trees (same number of trees in the global and local step).

show that they are still correctly detected, meaning the localization may not be perfect but still reasonably overlapping the ground truth.

4.4. Second experiment

In this second experiment, we show the interest of using the confidence maps in the cascade approach and compare our results with existing methods. We evaluate the methods of Cuingnet et al. (2012) and Criminisi et al. (2013) on our database. We gathered all the implementation details that we could find in their articles. If we could not find any parameter detail we used the same as ours. For the three methods we work at the same resolution (3 mm). The main differences between our implementation and the one of Criminisi et al. (2013) are: (i) they store histograms in the leaves, (ii) the final prediction is computed from the leaves with less uncertainty and by finding histogram maxima, (iii) the tree depth is 12, (iv) the minimum node size is 25. The differences between our implementation and the one of Cuingnet et al. (2012) are: (i) the voting voxels selection in the second step (ROI around the bounding box centers), (ii) the tree depth is 15, (iii) the minimum node size is 100, (iv) the bounding box centers only are re-estimated in the second regression step. In comparison our implementation has the following characteristics: (i) Gaussian distributions are stored in the leaves, (ii) the final prediction is the median over the votes with less uncertainty, (iii) the tree depth is 14, (iv) the voting voxels of the second step are selected thanks to the confidence maps, (v) all the bounding box parameters are re-estimated in the cascade, (vi) the minimum node size is 50.

The results are computed considering the bounding box wall distances (our results are computed by taking the median bounding box over the K_l best predictions). The figures are given in Table 3.

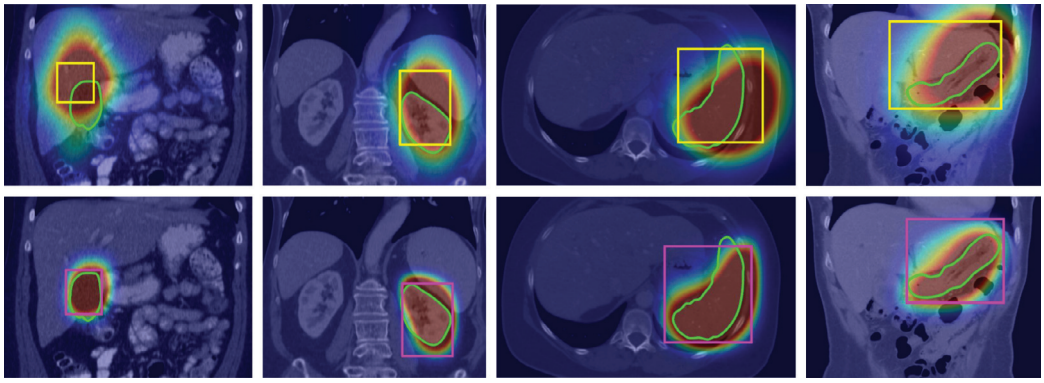

Fig. 13. Results after the global step (first row) and the local step (second row). The confidence maps and the median boxes are overlaid on the images, the green contour corresponds to the organ manual segmentation. From left to right, organs are in this order: gallbladder, left kidney, spleen and stomach. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

First of all, the performances after the global step (third line) show the benefit of the parameter optimization, as these results reach better accuracy than the results presented by Criminisi et al. (2013). Then we propose to compare the way of selecting the test voxels for the second local step:

- method 1: from the predicted boxes after the global step,
- method 2: from the thresholded confidence map.

The corresponding results are given in the last two lines of Table 3. The results after the method 1 first show the benefit of the cascade approach (additional iterations did not show significant improvements of the results), compared to the single step one. Fig. 13 confirms visually the benefit of the cascade approach. The results after the method 2 show the difference between our method and the original cascade approach. Significant improvements are especially noticed for the liver (gain of 1 mm, about 10% improvement) and the spleen (gain of 0.8 mm, about 10% improvement) whose shapes are very far from the bounding box approximation. This confirms the benefit of the shape prior for such organs. Moreover, a comparison of our results to those of Cuingnet et al. (2012) for the kidneys shows that the cascade approach is scalable and the increasing number of organs does not degrade the performances. Fig. 12 shows the results of method 1 and method 2 for a varying number of trees in the forest. We observe a very fast convergence of the accuracy, forests with three trees already giving very good results.

5. Conclusion

In this article we proposed a fast, robust and accurate method for the localization of multiple organs. We extended the idea of cascade of random forest regressors while introducing the concept of confidence map, which models the vote distributions with the addition of shape prior. We showed that the confidence map, with a proposed fast implementation, can enhance the consistency and accuracy of multi-organ localization for a limited computational overhead. It is a generic tool with promising potential, which can be used with any type of regressor and which is adaptable to different modalities (e.g. CT, MRI). Moreover its fuzziness property may be useful in many types of clinical applications, such as segmentation (for initialization) or visualization (to target the structures of interest for 3D rendering) for instance. It has been recently successfully applied in an automatic segmentation framework (Gauriau et al., 2015). The confidence maps are used for both the template initialization and the computation of organ-dedicated image forces.

This work also focused on parameters optimization and features understanding. The use of random forest is here very relevant as there is no need of a fine-tuning of the parameters. On the other hand a coarse optimization of the parameters helps improving the localization results. Our parameter study also highlights the link between the image information level and the range of features. We hope that we have given enough implementation details to make it accessible to any other researcher who would like to perform anatomy localization.

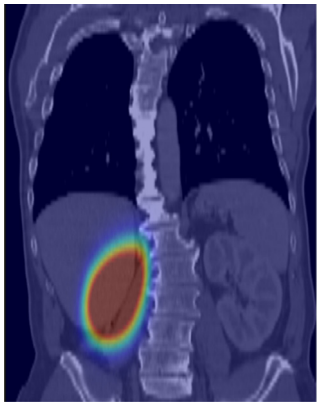

Fig. 14. Example of localization in the case of a missing right kidney.

Finally, the consistency and accuracy of this method may still be improved with the use of multiple probabilistic atlases per organ and with the addition of the rotation parameters. In its current form our method does not handle the missing organ case. Even if an organ is missing, the algorithm will give a localization prediction. In Fig. 14 we present an example of localization in a CT image with a missing right kidney. The result is plausible (the kidney is located below the liver) even if the organ is not present. As mentioned in Criminisi et al. (2013), the organ could be declared present if the confidence of the prediction is above a given threshold. Automatic detection of missing organs will be explored in a future work. Moreover, even though our results show consistent confidence maps in practice (not too far from the original organ shape), this is not guaranteed by our framework. We are currently investigating approaches to enforce the shape consistency of the final result. Keeping this method simple, with a low complexity and as generic as possible already gives reasonable results and makes it a fast and powerful component, that can be easily integrated into more advanced workflows.

Acknowledgment

This work was supported in part by an ANRT grant (008512012).

$C \leftarrow 0, \boldsymbol{d}_{\max} \leftarrow \mathbf{0}, d_{min} \leftarrow \max$;
for $k \leftarrow 1$ **to** K **do**
  $[b_{min}, b_{max}, \alpha_k] \leftarrow R_\theta[v_k]$;
  $c_k \leftarrow \frac{1}{2}(b_{min} + b_{max})$;
  $d_k \leftarrow b_{max} - b_{min}$;
  $d_{max} \leftarrow \texttt{max}(d_{max}, d_k)$;
  $d_{min} \leftarrow \texttt{min}(d_{min}, d_k)$;
$\hat{d} \leftarrow \texttt{quantize}(d_{max}, d_{min}J)$;
foreach $\hat{d}_j \in \hat{d}$ **do**
  $indices \leftarrow \texttt{getIndicesOfVotes}(\hat{d}_j, \hat{d})$;
  $C_j \leftarrow 0$;
  foreach $ind \in indices$ **do**
    $C_j[c_{ind}] \leftarrow C_j[c_{ind}] + \alpha_{ind}$;
  $A_j \leftarrow \texttt{rescaleAtlas}(A, \hat{d}_j)$;
  $C_j \leftarrow \texttt{convolve}(C_j, A_j)$;
  $C \leftarrow C + C_j$;

Algorithm 1: Pseudo-code for fast confidence map computation.

Appndix A

We present here the pseudo-code for the fast confidence map computation (Algorithm 1). The variables $\mathbf{b}_{\min}$, $\mathbf{b}_{\max}$, $\mathbf{d}_{\min}$, $\mathbf{d}_{\max}$, $\mathbf{c}_k$, $\mathbf{d}_k$ and $\mathbf{x}$ are in $\mathbb{R}^3$. The variable $\hat{\mathbf{d}}$ denotes the set of discretized size samples. The function `getBoxDimensions` computes the size of the box defined with parameters $\mathbf{b}_{\min}$ and $\mathbf{b}_{\max}$. The function `quantize` computes the different sample sizes from the range of boxes sizes. The function `getIndicesOfVotes` retrieves the indices of votes corresponding to the sample size $\hat{\boldsymbol{d}}_j$. The function `rescaleAtlas` rescales the atlas A anisotropically to fit to dimensions d.

References

Amit, Y., Geman, D., 1994. Randomized Inquiries about Shape; an Application to Handwritten Digit Recognition. Technical report 401. Department of Statistics, University of Chicago.

Amit, Y., Geman, D., 1997. Shape quantization and recognition with randomized trees. Neural Comput. 9 (7), 1545–1588.

Andonova, S., Elisseeff, A., Evgeniou, T., Pontil, M., 2002. A simple algorithm for learning stable machines. In: European Conference on Artificial Intelligence (ECAI), pp. 513–517.

Andriole, K.P., Wolfe, J.M., Khorasani, R., Treves, S.T., Getty, D.J., Jacobson, F.L., Steigner, M.L., Pan, J.J., Sitek, A., Seltzer, S.E., 2011. Optimizing analysis, visualization, and navigation of large image data sets: one 5000-section CT scan can ruin your whole day. Radiology 259 (2), 346–362.

Bishop, C.M., Lasserre, J., 2007. Generative or discriminative? Getting the best of both worlds. Bayesian Stat. 8, 3–24.

Blezek, D., Miller, J., 2007. Atlas stratification. Med. Image Anal. 11, 443–457.

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

Cabezas, M., Oliver, A., Lladó, X., Freixenet, J., Bach Cuadra, M., 2011. A review of atlas-based segmentation for magnetic resonance brain images. Comput. Methods Programs Biomed. 104 (3), e158–e177.

Chen, C., Zheng, G., 2014. Fully automatic segmentation of AP pelvis X-rays via random forest regression with efficient feature selection and hierarchical sparse shape composition. Comput. Vis. Image Understand. 126, 1–10.

Criminisi, A., Robertson, D., Konukoglu, E., Shotton, J., Pathak, S., White, S., Siddiqui, K., 2013. Regression forests for efficient anatomy detection and localization in computed tomography scans. Med. Image Anal. 17 (8), 1293–1303.

Criminisi, A., Shotton, J., 2013. Decision Forests for Computer Vision and Medical Image Analysis. Springer, London, New York.

Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E., 2011. Regression forests for efficient anatomy detection and localization in CT studies. In: Menze, B., Langs, G., Tu, Z., Criminisi, A. (Eds.), Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging. LNCS, vol. 6533. Springer, Berlin, Heidelberg, pp. 106–117.

Cuingnet, R., Prevost, R., Lesage, D., Cohen, L., Mory, B., Ardon, R., 2012. Automatic detection and segmentation of kidneys in 3D CT images using random forests. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (Eds.), Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 7512. Springer, Berlin, Heidelberg, pp. 66–74.

Dantone, M., Gall, J., Fanelli, G., Van Gool, L., 2012. Real-time facial feature detection using conditional regression forests. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2578–2585.

Dollár, P., Welinder, P., Perona, P., 2010. Cascaded pose regression. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1078–1085.

Fenchel, M., Thesen, S., Schilling, A., 2008. Automatic labeling of anatomical structures in MR FastView images using a statistical atlas. In: Metaxas, D., Axel, L., Fichtinger, G., SzÈ¨kely, G. (Eds.), Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 5241. Springer, Berlin, Heidelberg, pp. 576–584.

Fouquier, G., Atif, J., Bloch, I., 2012. Sequential model-based segmentation and recognition of image structures driven by visual features and spatial relations. Comput. Vis. Image Understand. 116 (1), 146–165.

Gall, J., Lempitsky, V., 2009. Class-specific hough forest for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1022–1029.

Gauriau, R., Ardon, R., Lesage, D., Bloch, I., 2015. Multiple template deformation. Application to abdominal organ segmentation. International Symposium on Biomedical Imaging (ISBI). New York, USA.

Gauriau, R., Cuingnet, R., Lesage, D., Bloch, I., 2014. Multi-organ localization combining global-to-local regression and confidence maps. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (Eds.), Medical Image Computing and Computer-Assisted Intervention (MICCAI). Springer, Berlin, Heidelberg, pp. 337–344.

Gauriau, R., Cuingnet, R., Prevost, R., Mory, B., Ardon, R., Lesage, D., Bloch, I., 2013. A generic, robust and fully-automatic workflow for 3D CT liver segmentation. In: Yoshida, H., Warfield, S., Vannier, M. (Eds.), Abdominal Imaging. Computation and Clinical Applications. LNCS, vol. 8198. Springer, Berlin, Heidelberg, pp. 241–250.

Glocker, B., Pauly, O., Konukoglu, E., Criminisi, A., 2012. Joint classification-regression forests for spatially structured multi-object segmentation. In: European Conference on Computer Vision (ECCV). Springer-Verlag, pp. 870–881.

Grahn, H.a., Lavesson, N., Lapajne, M.H., Slat, D., 2011. Cudarf: a cuda-based implementation of random forests. In: 9th ACS International Conference on Computer Systems and Applications (AICCSA). IEEE, pp. 95–101.

Hastie, T., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. (Springer series in statistics), 2nd Edition, Springer, New York, NY.

Ho, T.K., 1995. Random decision forests. In: IEEE Conference on Document Analysis and Recognition, vol. 1, pp. 278–282.

Ho, T.K., 1998. The random subspace method for constructing decision forests. Pattern Anal. Mach. Intell. 20 (8), 832–844.

Hoecker, A., Speckmayer, P., Stelzer, J., Therhaag, J., von Toerne, E., Voss, H., 2007. TMVA: Toolkit for Multivariate Data Analysis. PoS ACAT 040.

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

Kalinic, H., 2008. Atlas-based Image Segmentation: A Survey. Technical report. Department of Electronic Systems and Information Processing, Universiy of Zagreb.

Kontschieder, P., Bulò, S.R., Criminisi, A., Kohli, P., Pelillo, M., Bischof, H., 2012. Context-sensitive decision forests for object detection. In: Advances in Neural Information Processing Systems, pp. 431–439.

Kontschieder, P., Kohli, P., Shotton, J., Criminisi, A., 2013. Geof: geodesic forests for learning coupled predictors. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 65–72.

Konukoglu, E., Glocker, B., Zikic, D., Criminisi, A., 2013. Neighbourhood approximation using randomized forests. Med. Image Anal. 17 (7), 790–804.

Lay, N., Birkbeck, N., Zhang, J., Zhou, S., 2013. Rapid multi-organ segmentation using context integration and discriminative models. In: Gee, J.C., Joshi, S., Pohl, K.M., Wells, W.M., Zöllei, L. (Eds.), Information Processing in Medical Imaging (IPMI). LNCS, vol. 7917. Springer, Berlin, Heidelberg, pp. 450–462.

Liu, D., Zhou, K.S., Bernhardt, D., Comaniciu, D., 2010. Search strategies for multiple landmark detection by submodular maximization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2831–2838.

Liu, X., Song, Q., Mendonca, P., Tao, X., Bhotika, R., 2011. Organ labeling using anatomical model-driven global optimization. In: First IEEE International Conference on Healthcare Informatics, Imaging and Systems Biology (HISB), pp. 338–345.

Lombaert, H., Zikic, D., Criminisi, A., Ayache, N., Criminisi, A., Zikic, D., 2014. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (Eds.). Laplacian forests: semantic image segmentation by guided bagging.

Lu, C., Zheng, Y., Birkbeck, N., Zhang, J., Kohlberger, T., Tietjen, C., Boettger, T., Duncan, J., Zhou, S., 2012. Precise segmentation of multiple organs in CT volumes using learning-based approach and information theory. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (Eds.), Medical Image Computing and Computer-Assisted Intervention (MICCAI). LNCS, vol. 7511. Springer, Berlin, Heidelberg, pp. 462–469.

Montillo, A., Shotton, J., Winn, J., Iglesias, J., Metaxas, D., Criminisi, A., 2011. Entangled decision forests and their application for semantic segmentation of CT images. In: Information Processing in Medical Imaging (IPMI), pp. 184–196.

Morgan, J.N., Sonquist, J.A., 1963. Problems in the analysis of survey data, and a proposal. J. Am. Stat. Assoc. 58 (302), 415–434.

Murthy, R., Nunez, R., Szklaruk, J., Erwin, W., Madoff, D.C., Gupta, S., Ahrar, K., Wallace, M.J., Cohen, A., Coldwell, D.M., Kennedy, A.S., Hicks, M.E., 2005. Yttrium-90 microsphere therapy for hepatic malignancy: devices, indications, technical considerations, and potential complications. Radiographics 25 (Suppl._1), S41–S55.

Nempont, O., Atif, J., Bloch, I., 2013. A constraint propagation approach to structural model based image segmentation and recognition. Inform. Sci. 246, 1–27.

Ng, A.Y., Jordan, M.I., 2001. On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (Eds.), Advances in Neural Information Processing Systems 14. MIT Press, pp. 841–848.

Ozuysal, M., Fua, P., Lepetit, V., 2007. Fast keypoint recognition in ten lines of code. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Pauly, O., Glocker, B., Criminisi, A., Mateus, D., Möller, A., Nekolla, S., Navab, N., 2011. Fast multiple organ detection and localization in whole-body mr dixon sequences. In: Fichtinger, G., Martel, A., Peters, T. (Eds.), Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 6893. Springer, Berlin, Heidelberg, pp. 239–247.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Seifert, S., Barbu, A., Zhou, S.K., Liu, D., Feulner, J., Huber, M., Suehling, M., Cavallaro, A., Comaniciu, D., 2009. Hierarchical parsing and semantic navigation of full body ct data. SPIE Conf. Med. Imag. 7259, 725902.

Seifert, S., Kelm, M., Moeller, M., Mukherjee, S., Cavallaro, A., Huber, M., Comaniciu, D., 2010. Semantic annotation of medical images. In: Liu, B.J., Boonn, W.W. (Eds.), SPIE Conference on Medical Imaging, pp. 762808–762808–8.

Sharp, T., 2008. Implementing decision trees and forests on a GPU. In: Forsyth, D., Torr, P., Zisserman, A. (Eds.), European Conference on Computer Vision (ECCV). Springer, Berlin, Heidelberg, pp. 595–608.

Shimizu, A., Ohno, R., Ikegami, T., Kobatake, H., Nawano, S., Smutek, D., 2005. Simultaneous extraction of multiple organs from abdominal CT. In: Computer Assisted Radiology and Surgery (CARS), p. 1275.

Shotton, J., Johnson, M., Cipolla, R., 2008. Semantic texton forests for image categorization and segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Shotton, J., Winn, J., Rother, C., Criminisi, A., 2009. Textonboost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context. Int. J. Comput. Vis. 81 (1), 2–23.

Sofka, M., Zhang, J., Zhou, S.K., Comaniciu, D., 2010. Multiple object detection by sequential monte carlo and hierarchical detection network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1735–1742.

Sun, M., Kohli, P., Shotton, J., 2012. Conditional regression forests for human pose estimation. In: Computer Vision and Pattern Recognition (CVPR), pp. 3394–3401.

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

Tu, Z., 2008. Auto-context and its application to high-level vision tasks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

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

Wolz, R., Chu, C., Misawa, K., Mori, K., Rueckert, D., 2012. Multi-organ abdominal CT segmentation using hierarchically weighted subject-specific atlases. Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 7510, pp. 10–17.

Wu, D., Sofka, M., Birkbeck, N., Zhou, S.K., 2014. Segmentation of multiple knee bones from CT for orthopedic knee surgery planning. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (Eds.), Medical Image Computing and Computer-Assisted Intervention (MICCAI). LNCS, vol. 8673. Springer International Publishing, pp. 372–380.

Zhan, Y., Zhou, X.S., Peng, Z., Krishnan, A., 2008. Active scheduling of organ detection and segmentation in whole-body medical images. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (Eds.), Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 5241. Springer, Berlin, Heidelberg, pp. 313–321.

Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., Comaniciu, D., 2007. Fast automatic heart chamber segmentation from 3d ct data using marginal space learning and steerable features. In: IEEE International Conference on Computer Vision (ICCV), pp. 1–8.

Zhou, K.S., 2014. Discriminative anatomy detection: classification vs regression. Pattern Recog. Lett. 43, 25–38.

Zhou, X.S., Peng, Z., Zhan, Y., Dewan, M., Jian, B., Krishnan, A., Tao, Y., Harder, M., Grosskopf, S., Feuerlein, U., 2010. Redundancy, redundancy, redundancy: the three keys to highly robust anatomical parsing in medical images. In: Proceedings of the International Conference on Multimedia Information Retrieval. ACM, pp. 175–184.

Zikic, D., Glocker, B., Criminisi, A., 2013. Atlas encoding by randomized forests for efficient label propagation. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (Eds.), Medical Image Computing and Computer-assisted Intervention (MICCAI). LNCS, vol. 8151. Springer, Berlin, Heidelberg, pp. 66–73.