Validating Automatic Semantic Annotation of Anatomy in DICOM CT Images

Sayan D. Pathak[a], Antonio Criminisi[b], Jamie Shotton[b], Steve White[a], Duncan Robertson[b], Bobbi Sparks[a], Indeera Munasinghe[b] and Khan Siddiqui[a]

[a]Microsoft Health Solutions Group R&D, 1 Microsoft Way, Redmond WA, USA 98052
[b]Microsoft Research Labs, JJ Thomson Ave, Cambridge, Cambridgeshire, UK CB3 0FB

ABSTRACT

In the current health-care environment, the time available for physicians to browse patients' scans is shrinking due to the rapid increase in the sheer number of images. This is further aggravated by mounting pressure to become more productive in the face of decreasing reimbursement. Hence, there is an urgent need to deliver technology which enables faster and effortless navigation through sub-volume image visualizations. Annotating image regions with semantic labels such as those derived from the RADLEX ontology can vastly enhance image navigation and sub-volume visualization. This paper uses random regression forests for efficient, automatic detection and localization of anatomical structures within DICOM 3D CT scans. A regression forest is a collection of decision trees which are trained to achieve direct mapping from voxels to organ location and size in a single pass. This paper focuses on comparing automated labeling with expert-annotated ground-truth results on a database of 50 highly variable CT scans. Initial investigations show that regression forest derived localization errors are smaller and more robust than those achieved by state-of-the-art global registration approaches. The simplicity of the algorithm's context-rich visual features yield typical runtimes of less than 10 seconds for a 512^3 voxel DICOM CT series on a single-threaded, single-core machine running multiple trees; each tree taking less than a second. Furthermore, qualitative evaluation demonstrates that using the detected organs' locations as index into the image volume improves the efficiency of the navigational workflow in all the CT studies.

Keywords: DICOM, RADLEX, Semantic, Tagging, Classification

1. INTRODUCTION

Nowadays, physicians face ever increasing volumes of image studies to be interpreted in conjunction with decreasing reimbursement and pressure to become more productive. The reduced amount of time allocated to each patient yields the urgent need for technology to aid visual inspection and analysis of their scans. This paper describes and validates one such technology, based on automatic semantic labeling of DICOM CT images. Automatic localization of semantic components in images yields, among other things, efficient browsing of archived scans and automated linking of text-based clinical data with DICOM image content.

Popular approaches for the automated annotation of image regions typically involve registration of atlases (e.g., via the MedInria package[1]) or the application of a sequence of filters/classifiers[2,3] which have demonstrated limited robustness and are computationally intensive. This paper uses a multivariate random regression forest (RRF) algorithm for the efficient automatic detection and localization of anatomical structures within CT scans. Regression forests are similar to the more popular classification forests, but are trained to predict the continuous position of bounding boxes associated with the various organs/structures of interest. This probabilistic approach, based on the maximization of prediction confidence, enables direct mapping from voxels to organ location and size.[4,5] This paper demonstrates that automatic annotation of image regions with semantic labels – such as those derived from the RADLEX ontology is robust and streamlines navigation and visualization of large DICOM CT volumes. The high computational efficiency and lack of human intervention enable off-line image annotation either during data acquisition, data archival in PACS, or on demand. The annotations can then be used for efficient sub-volume retrieval, e.g., in PACS-based client - server rendering.

Further author information: (Send correspondence to S. P.)
S.P.: E-mail: FirstNamePA at microsoft.com, Telephone: 1 425 538 7386

The key contribution of this paper is evaluating the efficacy of our previously published core anatomy localization using RRF algorithm[5] when used for navigating through large DICOM CT scans. We primarily focus on the algorithm's accuracy when navigating towards an organ of choice, a key scenario for improving DICOM CT rendering over a client-server infrastructure. The goal of our study is to ensure that the regression forest algorithm will enable a user to navigate to an organ of interest: i) accurately, ii) quickly, iii) with minimal interaction, and iv) with low bandwidh between the client and server.

Outline. Section 2 summarizes the RRF based anatomy bounding box detection algorithm. Section 3 introduces the error measures used for bounding box aided navigational efficacy evaluation. Section 4 illustrates our various findings of both robustness of the organ detection algorithm as well as its ability to enable automated image navigation. Finally we summarize key insights in section 5.

2. ALGORITHM: HIERARCHICAL REGRESSION FOR ORGAN LOCALIZATION

This section summarizes the algorithm for the automatic localization of anatomy of interest in volumetric CT scans. For completeness this section summarizes our multi-organ localization algorithm. For a full explanation please refer to Criminisi et. al.[5]

2.1 Overview

The RRF algorithm estimates the position of the bounding box around an anatomical structure by pooling contributions from all voxels in a CT volume. This approach clusters voxels together based on their appearance, their spatial context and, above all, their confidence in predicting position and size of all anatomical structures.

The regression trees at the basis of the forest predictor are trained on a predefined set of volumes with associated ground-truth bounding boxes. The training process selects at each node the visual feature that maximizes the confidence on its prediction for a given structure. The tighter the predicted bounding box distribution, the more likely that feature is selected in a node of the tree.

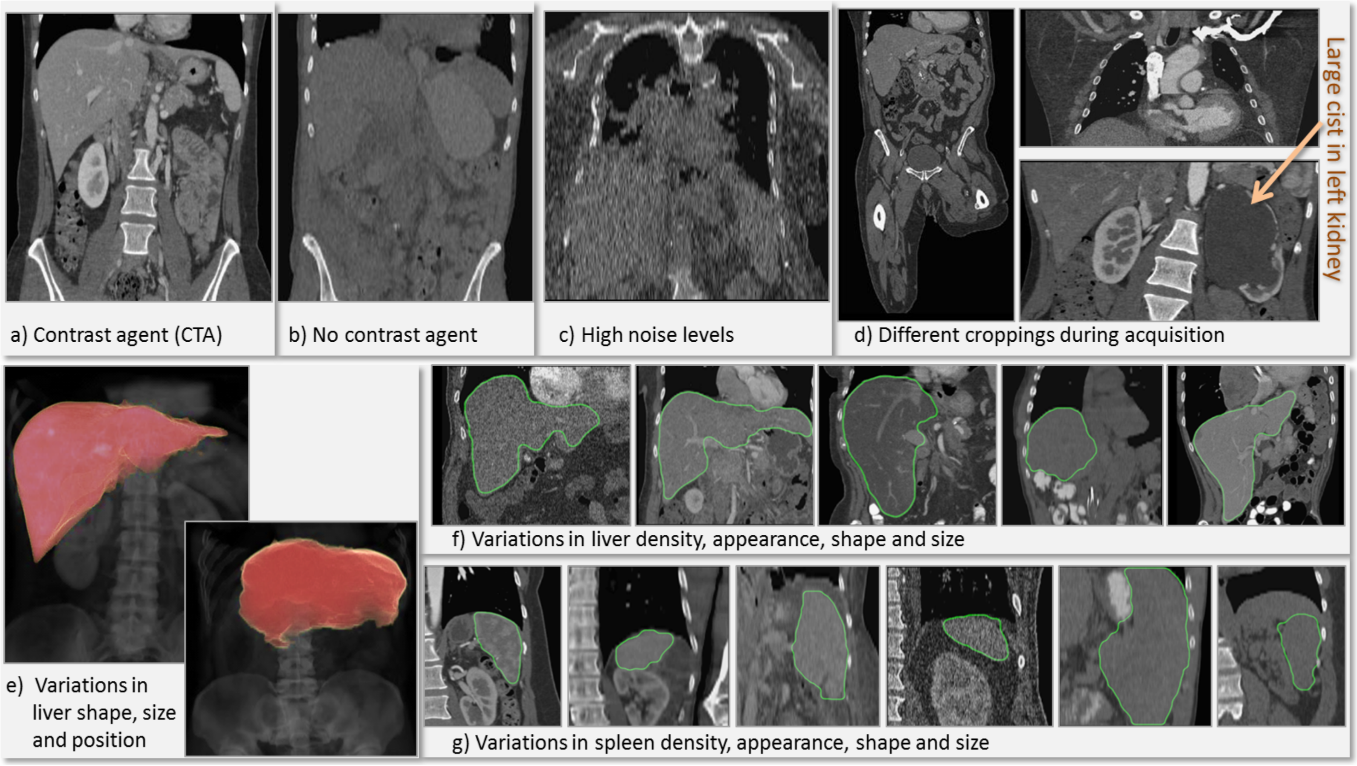
During the testing phase, voxels in an image volume are provided as an input to all the trees in the forest, simultaneously. At each node the corresponding visual test is applied to the voxel and based on the outcome the voxel is sent to the left or right child. When the voxel reaches a leaf node, the stored distribution is used as the probabilistic vote cast by the voxel itself. Only the leaves with highest localization confidence are used for the final estimation of each organs bounding box location. Please see Criminisi et. al.[5] for details.

This section presents our mathematical parameterization and the details of our multi-class regression forest with application to anatomy localization.

Mathematical notation. Vectors are represented in boldface (*e.g.* $\mathbf{v}$), matrices as teletype capitals (*e.g.* $\mathtt{\Lambda}$) and sets in calligraphic style (*e.g.* $\mathcal{S}$). E.g. the position of a voxel in a CT volume is denoted $\mathbf{v} = (v_x, v_y, v_z)$.

The labelled database. The anatomical structures we wish to recognize are $\mathcal{C}$ ={ `heart, liver, left lung, right lung, l. kidney, r. kidney, l. pelvis, r. pelvis, spleen`}. We are given a database of 100 scans comprising patients with different conditions and large differences in body size, pose, image cropping, resolution, scanner type, and possible use of contrast agents (fig. 1). All CT scans have been manually annotated with 3D bounding boxes tightly drawn around the structures of interest (see fig. 2a). The bounding box for an organ $c \in \mathcal{C}$ is parametrized as a 6-vector $\mathbf{b}_c = (b_c^{\mathtt{L}}, b_c^{\mathtt{R}}, b_c^{\mathtt{A}}, b_c^{\mathtt{P}}, b_c^{\mathtt{H}}, b_c^{\mathtt{F}})$ where each component represents the absolute position (in mm) of the corresponding axis-aligned wall[*]. Voxel sizes are $\sim 0.5 - 1.0mm$ along x and y, and $\sim 1.0 - 5.0mm$ along z. The images have not been pre-registered or normalized in any way. The goal is to localize anatomic structures of interest accurately and automatically, despite such large variability. Next we describe how this is achieved.

[*] Superscripts follow standard radiological orientation convention: $\mathtt{L}$ = left, $\mathtt{R}$ = right, $\mathtt{A}$ = anterior, $\mathtt{P}$ = posterior, $\mathtt{H}$ = head, $\mathtt{F}$ = foot.


a) Contrast agent (CTA)

b) No contrast agent

c) High noise levels

d) Different croppings during acquisition

Large cist in left kidney

e) Variations in liver shape, size and position

f) Variations in liver density, appearance, shape and size

g) Variations in spleen density, appearance, shape and size

Figure 1. **Variability in our labelled database. (a,b,c)** Variability in appearance due to presence of contrast agent, or noise. **(d)** Difference in image geometry due to acquisition parameters and possible anomalies. **(e)** Volumetric renderings of liver and spine to illustrate large changes in their relative position and in the liver shape. **(f,g)** Mid-coronal views of liver and spleen across different scans in our database to illustrate their variability. All views are metrically and photometrically normalized to aid comparison.

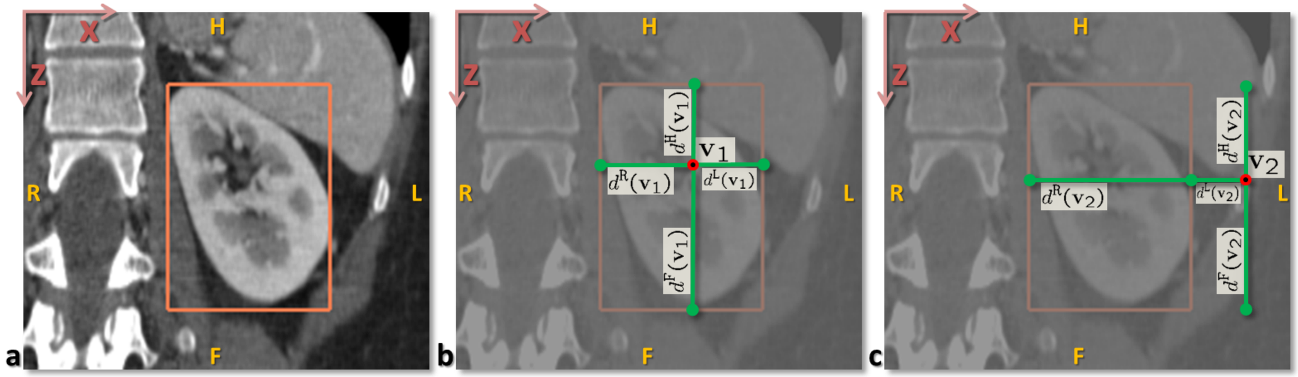

a
b
c
X
Z
H
R
L
F
$\mathbf{v}_1$
$d^{\mathrm{H}}(\mathbf{v}_1)$
$d^{\mathrm{R}}(\mathbf{v}_1)$
$d^{\mathrm{L}}(\mathbf{v}_1)$
$d^{\mathrm{F}}(\mathbf{v}_1)$
$\mathbf{v}_2$
$d^{\mathrm{H}}(\mathbf{v}_2)$
$d^{\mathrm{R}}(\mathbf{v}_2)$
$d^{\mathrm{L}}(\mathbf{v}_2)$
$d^{\mathrm{F}}(\mathbf{v}_2)$

Figure 2. **Problem parametrization. (a)** A coronal view of a left kidney and the associated ground-truth bounding box (in orange). **(b,c)** *Every* voxel $\mathbf{v}_i$ in the volume votes for the position of the six walls of each organ's 3D bounding box via 6 relative, offset displacements $d^{\mathrm{k}}(\mathbf{v}_i)$ in the three orthogonal planes along x, y and z axes.

2.2 Problem parametrization and regression forest learning

Key to the algorithm is the fact that *all* voxels in a test CT volume contribute with varying confidence to estimating the position of the six walls of *all* structures' bounding boxes (see fig. 2b,c). Intuitively, some distinct voxel clusters (*e.g.* ribs or vertebrae) may predict the position of an organ (*e.g.* the heart) with high confidence. Thus, during testing, those clusters will be used as reference (landmarks) for the localization of those anatomical structures. Our aim is to learn to cluster voxels together based on their appearance, their spatial context and their confidence in predicting position and size of all anatomical structures. We tackle this simultaneous feature selection and parameter regression task with a multi-class random regression forest (see fig. 3).

2.2.1 Forest training

The training process constructs each regression tree and decides at each node how to best split the incoming voxels. We are given a subset of all labelled CT volumes (the training set), and the associated ground-truth organ bounding boxes (fig. 2a). The size of the forest T is fixed and all trees are trained in parallel. Each voxel is pushed through each of the trees starting at the root. Each split node applies the following binary test $\xi_j > f(\mathbf{v}; \boldsymbol{\theta}_j) > \tau_j$ and based on the result sends the voxel to the left or right child node. $f(.)$ denotes the feature response computed for the voxel $\mathbf{v}$. The parameters $\boldsymbol{\theta}_j$ represent the visual feature which applies to the j^{th} node. Our visual features are similar to those in,[5–7] *i.e.* mean intensities over displaced, asymmetric cuboidal regions. These features are efficient and capture spatial context. The feature response is $f(\mathbf{v}; \boldsymbol{\theta}_j) = |F_1|^{-1} \sum_{\mathbf{q} \in F_1} I(\mathbf{q}) - |F_2|^{-1} \sum_{\mathbf{q} \in F_2} I(\mathbf{q})$; with F_i indicating 3D box regions and I the intensity. F_2 can be the empty set for unary features. Randomness is injected by making available at each node only a random sample of all features. This technique has been shown to increase the generalization of tree-based predictors.[4] Next we discuss how to optimize each node.

Node optimization. Each voxel $\mathbf{v}$ in each training volume is associated with an offset $\mathbf{d}_c(\mathbf{v})$ with respect to the bounding box $\mathbf{b}_c$ for each class $c \in \mathcal{C}$ (see fig. 2b,c). Such offset is denoted: $\mathbf{d}_c(\mathbf{v}) = (d_c^{\mathtt{L}}, d_c^{\mathtt{R}}, d_c^{\mathtt{A}}, d_c^{\mathtt{P}}, d_c^{\mathtt{H}}, d_c^{\mathtt{F}}) \in R^6$, with $\mathbf{b}_c(\mathbf{v}) = \hat{\mathbf{v}} - \mathbf{d}_c(\mathbf{v})$ and $\hat{\mathbf{v}} = (v_x, v_x, v_y, v_y, v_z, v_z)$. As with classification, node optimization is driven by maximizing an information gain measure, defined as: $IG = H(\mathcal{S}) - \sum_{i=\{\mathtt{L},\mathtt{R}\}} \omega_i H(\mathcal{S}_i)$ where H denotes entropy, $\mathcal{S}$ is the set of training points reaching a node and $\mathtt{L}, \mathtt{R}$ denote the left and right children. In classification the entropy is defined over distributions of discrete class labels. In regression instead we measure the purity of the probability density of the real-valued predictions. For a single class c we model the distribution of the vector $\mathbf{d}_c$ at each node as a multivariate Gaussian; *i.e.* $p(\mathbf{d}_c) = \mathcal{N}(\mathbf{d}_c; \overline{\mathbf{d}}_c, \Lambda_c)$, with the matrix Λ_c encoding the covariance of $\mathbf{d}_c$ for all points in $\mathcal{S}$. The differential entropy of a multivariate Gaussian can be shown to be $H(\mathcal{S}) = \frac{n}{2}(1 + \log(2\pi)) + \frac{1}{2} \log |\Lambda_c(\mathcal{S})|$ with n the number of dimensions ($n = 6$ in our case). Algebraic manipulation yields the following regression information gain: $IG = \log |\Lambda_c(\mathcal{S})| - \sum_{i=\{\mathtt{L},\mathtt{R}\}} \omega_i \log |\Lambda_c(\mathcal{S}_i)|$. In order to handle simultaneously all $|\mathcal{C}| = 9$ anatomical structures the information gain is adapted to: $IG = \sum_{c \in \mathcal{C}} \left(\log |\Lambda_c(\mathcal{S})| - \sum_{i=\{\mathtt{L},\mathtt{R}\}} \omega_i \log |\Lambda_c(\mathcal{S}_i)| \right)$ which is readily rewritten as

$$IG = \log |\Gamma(\mathcal{S})| - \sum_{i=\{\mathtt{L},\mathtt{R}\}} \omega_i \log |\Gamma(\mathcal{S}_i)|, \quad \text{with } \Gamma = \mathrm{diag}\left(\Lambda_1, \cdots, \Lambda_c, \cdots, \Lambda_{|\mathcal{C}|}\right). \tag{1}$$

Maximizing (1) encourages minimizing the determinant of the $6|\mathcal{C}| \times 6|\mathcal{C}|$ covariance matrix Γ, thus decreasing the uncertainty in the probabilistic vote cast by each cluster of voxels on each organ pose. Node growing stops when IG is below a fixed threshold, too few points reach the node or a maximum tree depth D is reached (here $D = 7$). After training, the j^{th} split node remains associated with the feature $\boldsymbol{\theta}_j$ and thresholds ξ_j, τ_j. At each leaf node we store the learned mean $\overline{\mathbf{d}}$ (with $\mathbf{d} = (\mathbf{d}_1, \cdots, \mathbf{d}_c, \cdots, \mathbf{d}_{|\mathcal{C}|})$) and covariance Γ, (fig. 3b).

2.2.2 Forest testing

Given a previously unseen CT volume $\mathcal{V}$, each voxel $\mathbf{v} \in \mathcal{V}$ is pushed through each tree starting at the root and the corresponding sequence of tests applied (see fig. 3). The voxel stops when it reaches its leaf node $l(\mathbf{v})$, with l indexing leaves across the whole forest. The stored distribution $p(\mathbf{d}_c|l) = \mathcal{N}(\mathbf{d}_c; \overline{\mathbf{d}}_c, \Lambda_c)$ for class c also defines the posterior for the absolute bounding box position: $p(\mathbf{b}_c|l) = \mathcal{N}(\mathbf{b}_c; \overline{\mathbf{b}}_c, \Lambda_c)$, since $\overline{\mathbf{b}}_c(\mathbf{v}) = \hat{\mathbf{v}} - \overline{\mathbf{d}}_c(\mathbf{v})$. The posterior probability for $\mathbf{b}_c$ is now given by

$$p(\mathbf{b}_c) = \sum_{l \in \tilde{\mathcal{L}}} p(\mathbf{b}_c|l) p(l). \tag{2}$$

$\tilde{\mathcal{L}}$ is a subset of all forest leaves. Here we select $\tilde{\mathcal{L}}$ as the set of leaves which have the smallest uncertainty (for each class c) and contain 1% of all test voxels. Finally $p(l) = 1/|\tilde{\mathcal{L}}|$ if $l \in \tilde{\mathcal{L}}, 0$ otherwise. This is different from averaging the output of all trees (as done *e.g.* in[6,8]) as it uses the most confident leaves, independent from which tree in the forest they come from.

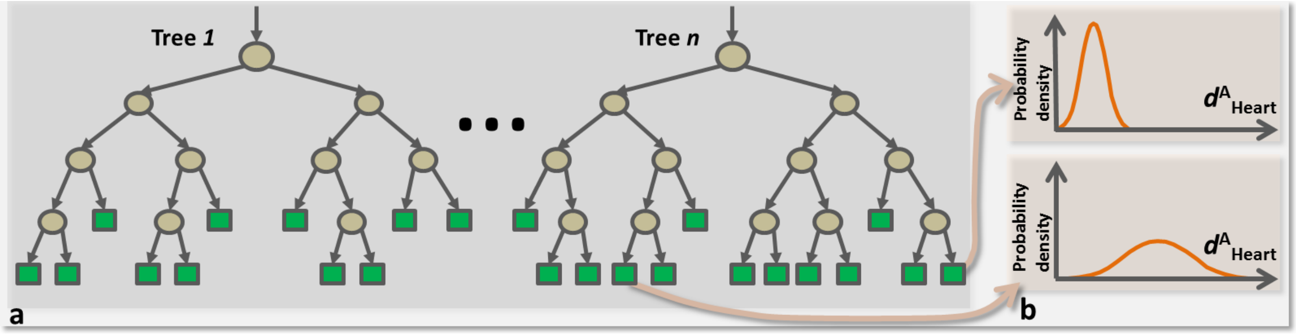


Figure 3. **A regression forest** is an ensemble of different regression trees. Each leaf contains a distribution for the continuous output variable/s. Leaves have associated different degrees of confidence (illustrated by the "peakiness" of distributions). During testing each text voxel is "pushed" through each tree starting at the root until it reaches a leaf node. The corresponding prediction is read at the leaves.

Anatomy detection. The organ c is declared present in the scan if $p(\mathbf{b}_c = \tilde{\mathbf{b}}_c) > \beta$, with $\beta = 0.5$.

Anatomy localization. The final prediction $\tilde{\mathbf{b}}_c$ for the absolute position of the c^{th} organ is given by the expectation $\tilde{\mathbf{b}}_c = \int_{\mathbf{b}_c} \mathbf{b}_c p(\mathbf{b}_c) d\mathbf{b}_c$.

3. VALIDATION AND VERIFICATION

This section evaluates the contribution of our algorithm in increasing the efficiency of navigating through CT images. We wish to evaluate speed and ease of interaction as well as accuracy of localization; and its practical implications in a radiological suite. In order to facilitate our evaluation, we have two different measures to compare the detected and the ground-truth organ bounding boxes. This section describes those measures and our efforts towards a quantitative evaluation of the navigational user experience.

Measure 1: Bounding wall prediction error. One of the outputs of our algorithm is the location of the bounding box with respect to the image volume. The intended use-case for this study leverages the detected bounding box to identify the image subvolume where the organ of interest is likely to be located. Hence, we compare the algorithmically detected bounding box wall with the ground truth bounding box. Ideally one would expect all the walls (6 walls in a 3D space namely, left/right, head/foot and anterior/posterior) to align with the ground truth bounding box. Fig. 4(a) illustrates the 4 different wall errors between the detected bounding box and the correspoding walls in the ground truth for a 2D schematic layout. The errors are defined as absolute difference between predicted and true wall positions.

Measure 2: Centroid-hit error. In our application, when the user desires to navigate to a certain organ, the application shall perform a Multi-Planar Rendering (MPR) of the image volume with the three cross-sectional planes centered at the centroid of the organ's detected bounding box. In order to measure whether the MPR view contains the selected organ we determine if the centroid of the detected bounding box falls within the ground-truth bounding box (schematically represented by fig. 4(b)). However, we expect that in some cases the detected centroid may lie outside the ground truth box. In that case we would like to measure the Cartesian error with respect to the x, y and z extents of the ground truth bounding box. Fig. 4(c) shows one such situation where the detected box is taller compared to the ground-truth bounding box. This leads to an error in the prediction along the vertical dimension even though the horizontal prediction falls within the ground-truth box. This measure enables us to evaluate our semantic navigation use case against a variety of datasets. As shown later it also helps gather specific insight into where the algorithm could be further improved.

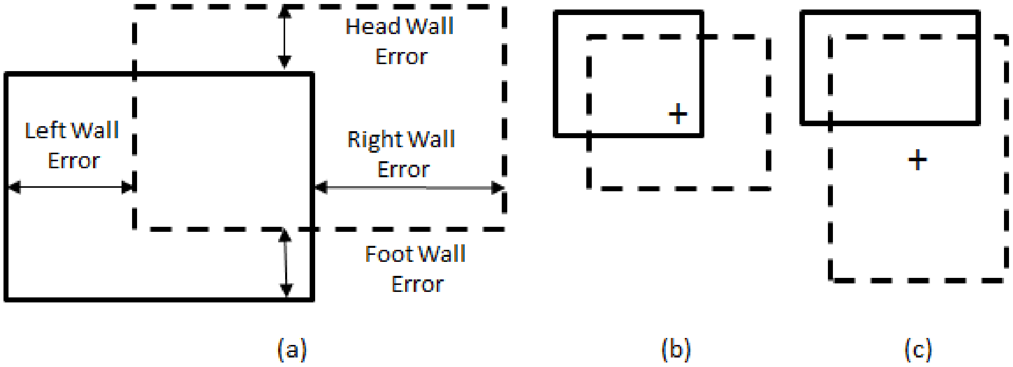


Figure 4. **Error measures.** **(a)** 2-D schematic depiction of the 4 errors associated with the position of each wall in the predicted bounding box (dotted line) as compared to ground-truth box (solid). **(b)** Centroid of the predicted bounding box falls inside the ground-truth bounding box. **(c)** Centroid of the predicted bounding box falls outside the ground-truth bounding box (solid).

4. RESULTS AND DISCUSSION

In this section, we first report the results from our comparison between atlas registration and RRF based organ detection and accuracy of the organ localization (section 4.1) followed by our evaluation of the user experience with the navigational enhancement facilitated by organ localization (section 4.2). For these three evaluations we use two separate datasets as described below.

- For the first two evaluations (reported in section 4.1) we use 100 DICOM CT images from a variety of different sources with focus on capturing large variability (e.g., amputees, missing kidney, collapsed lung, large skeletal deformations, body profiles, different scanners, z-axis resolution, with/without contrast etc.) in DICOM CT scans allowing us to build a very robust random forest. This is referred to **dataset A**.

- For the second evaluation (reported in section 4.2), our goal was to evaluate the experience one would expect when using our trained regression random forest (from dataset A) on DICOM CT scans derived from a different institution that is not represented in dataset A. For these we use 50 consecutive DICOM CT scans and refer to as **dataset B**.

4.1 Comparison with affine, atlas-based registration and accuracy evaluation

For this evaluation, using dataset A we trained the algorithm on 55 volumes and tested on the remaining 45 CT volumes. The bounding walls localization errors (measure 1) are summarized in table 1. The table aggregates results over all bounding box sides. Despite the large data variability we obtain a mean error of only $\sim 1.7cm$ (median $\sim 1.1cm$). On average, errors along the z direction are about twice as large as those in x and y. This is due both to reduced resolution and larger variability in cropping along the z direction. Consistently good results are obtained for different choices of training set as well as different training runs.

One key requirement of our use-cases (i.e., clinical) is that the algorithm generate results quickly. Thus, here we chose to compare our results with those obtained from a comparably fast atlas-based algorithm; one based on global registration. A reference atlas is selected from the training set as the volume which, when registered with all *test* scans, produced the minimum localization error. Registration was attained using the popular MedInria package.[1] We chose the global registration algorithm (from the many implemented) and associated parameters that produced best results on the *test* set. The algorithm that met these criteria was the block-matching with an affine transformation model. Note that optimizing the atlas selection and the registration algorithm on the test set produces results which are biased in favor of the atlas-based technique and yields a much tougher evaluation ground for our RRF algorithm.

The resulting atlas based errors (computed on the same test set) are summarized in table 1. They show much larger error mean and standard deviation (about double) than our random regression forest (RRF) based

organ	mean (Atlas)	std (Atlas)	median (Atlas)	mean (RRF)	std (RRF)	median (RRF)	mean (Diff)	std (Diff)	median (Diff)
heart	24.4	27.0	15.5	15.4	15.5	9.3	9.0	11.5	6.2
liver	34.2	59.3	16.4	17.1	16.5	13.2	17.1	42.8	3.2
left lung	27.8	29.9	15.7	17.0	17.2	11.3	10.8	12.7	4.4
right lung	27.0	27.6	18.0	15.6	16.3	10.6	11.4	11.3	7.4
left kidney	39.1	55.6	25.7	17.3	16.5	12.8	21.8	39.1	12.9
right kidney	28.3	53.3	15.4	18.5	18.0	12.3	9.8	35.3	3.1
left pelvis	23.4	43.3	10.9	13.2	14.0	8.8	10.2	29.3	2.1
right pelvis	22.4	43.5	11.8	12.8	13.9	8.4	9.6	29.6	3.4
spleen	36.0	57.2	20.1	20.7	22.8	12.9	15.3	34.4	7.2
Across all organs	31.3	50.0	17.2	16.7	17.0	11.5	14.6	33.0	5.7

Table 1. Bounding box localization errors (mean, standard dev and median) (in mm) from the Atlas based and RRF based localization. Last three colums show the difference between the Atlas and RRF based errors.

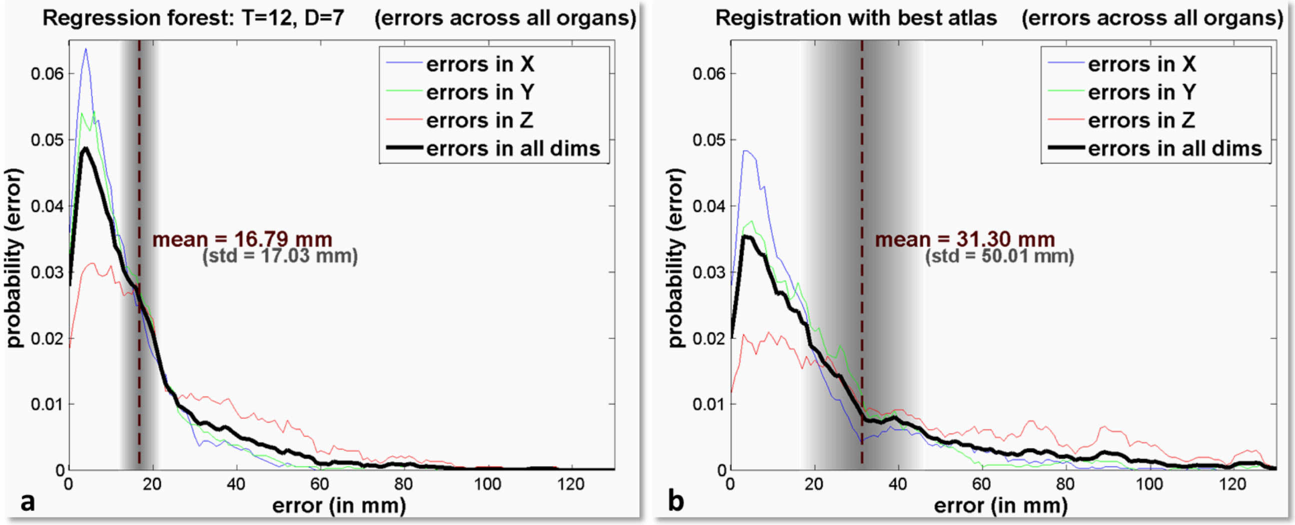


Figure 5. **Comparison with atlas-based registration.** Distributions of localization errors for **(a)** our algorithm, and **(b)** the atlas-based technique. The atlas-induced errors show more mass in the tails, which is reflected by a larger standard deviation (std). The width of the vertical shaded band is proportional to the standard deviation.

approach (see table 1). Note, the large positive difference between atlas and RRF bounding box localization errors. In terms of computational speed, atlas based registration is achieved in between 90s and 180s per scan, on the same dual-core machine (*cf.* our algorithm runtime is $\sim 6s$ for $T = 12$ trees).

Figure 5 further illustrates the difference in accuracy between the two approaches. In the registration case larger tails of the error distribution suggest a less robust behavior because larger errors are produced more often than in our algorithm. This is reflected in larger values of the error mean and standard deviation and is consistent with our visual inspection of the registrations. In fact, in $\sim 30\%$ cases the process got trapped in local minima and produced grossly inaccurate alignment. Those cases tend not to get improved when using a local registration step which doesn't help avoid getting trapped in local minima, while adding considerably to the runtime. Similar results have been obtained when comparing our results to the Elastix algorithm.[9]

Testing each tree on a typical 512^3 scan takes approximately 1s with our C++ implementation; and all trees may be tested in parallel. Further speed-ups can be achieved with more low-level code optimizations. A regression forest with 6 trees takes $\sim$ 10MB of memory. This is in contrast with the roughly 100MB taken by each atlas. The issue of model size and runtime efficiency may be exacerbated by the use of more accurate and costly multi-atlas techniques.[10] Finally, increasing the regression forest training set usually decreases the test error without affecting the test runtime, while in multi-atlas techniques increasing the number of atlases linearly increases the runtime latency.

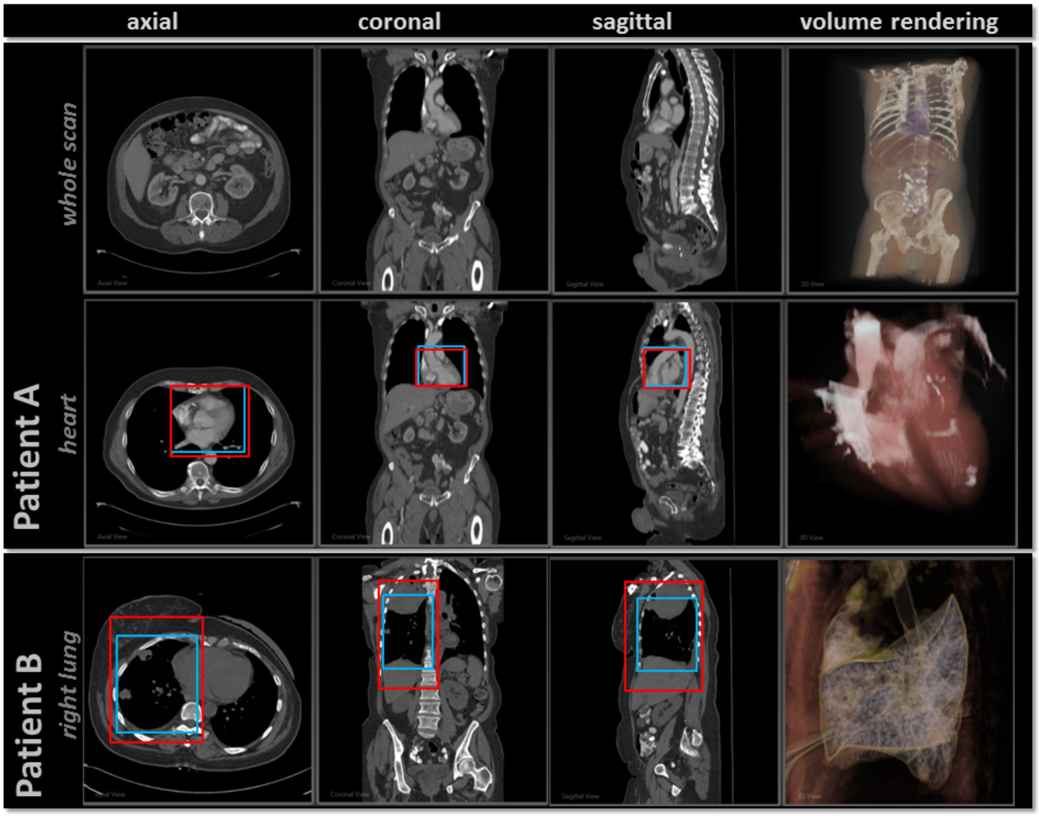


Figure 6. **Qualitative results.** Our application for single-click semantic navigation through two CT studies showing axial, coronal, sagittal and 3D views. Ground-truth organ bounding boxes are shown in blue. Automatically detected ones are shown in red. The detected bounding boxes are very close to the ground truth ones. Once each organ has been detected the 3D camera is repositioned, the appropriate cropping applied and the best colour transfer function automatically selected, thus saving the user much valuable time.

4.2 Improvement in navigation

In this section we evaluate the effect of our automatic organ localization algorithm on the visual navigation application. We have performed both qualitative and quantitative evaluation. We report results from using the Regression Forest derived from dataset A and applying that for organ detection in dataset B.

Using qualitative measure From a radiologist or a clinician's point of view one of the most useful applications of this technique is to optimize navigation of large volume CT datasets. It is more important to be in the area of the region of interest rather than accurately segment the region for fast navigation. Keeping this use case in mind, the qualitative assessment we established was to see if any portion of the organ of interest was within the bounding box in any of the planes. Qualitative results on automatic anatomy localization on whole-body CT scans are shown in fig. 6. In our manual evaluation of dataset B, we found in 9 out of 10 cases when the organ of interest was present in the scan, the algorithm allows rapid navigation to the region of interest.

Using wall prediction error. Fig. 7 shows the distribution of errors in the prediction of the six walls of the bounding box. Note that the over all, median error for most wall predictions is within 2 cm in L/R and A/P plane and below 5 cm in the Head/Foot plane. This implies that our algorithm typically navigates the user towards their chosen organ. Furthermore, the Head/Foot error plots illustrate the previously mentioned issues with cropping and z-axis resolution. The data show that, with the exception of the head (upper) position of the lung wall, the regression forest organ detector has the accuracy suitable for the navigational use case.

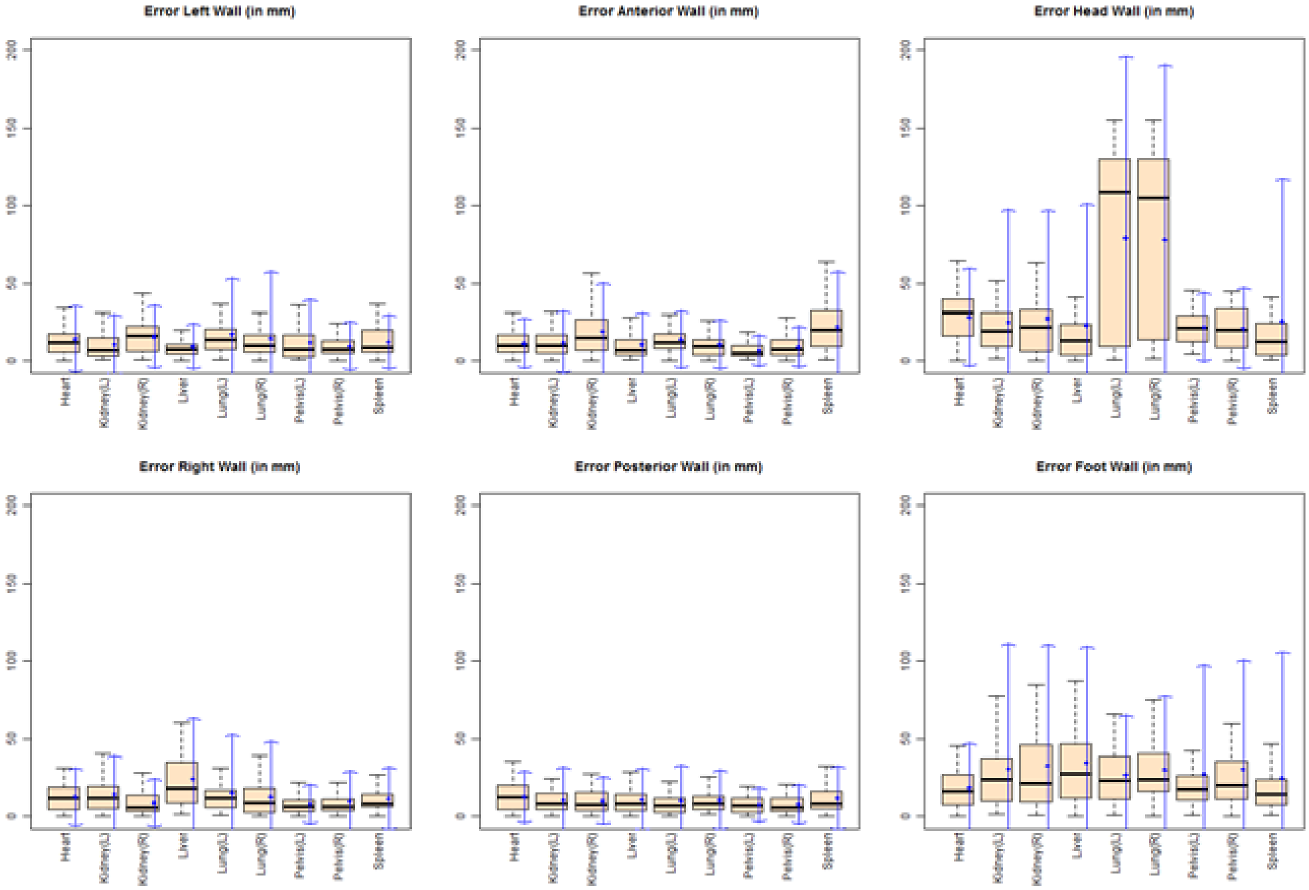


Figure 7. Boxplot of wall prediction error for all organs of interest. Along with the boxplots, the mean error measure (as a dot within the box) and an arrow indicating ±2 × standard deviation span are also shown. The bold horizontal bar within each box shows the median.

Additionally, as a measure of robustness of our wall detection algorithm, we report the numberof times the error in the detected wall fall within the 2 standard deviations of the mean wall estimation error for each organ. Table 2 shows the percentage number of wall predictions that fall within 2 standard deviations from the mean error. At least 94% of the wall predictions for all organs fall within the 2 standard deviation from the mean wall error, indicating a high level of robustness.

	Left	Right	Anterior	Posterior	Head	Foot
Heart	98	98	98	98	98	100
Kidney(L)	100	96	98	98	98	98
Kidney(R)	98	96	96	98	98	98
Liver	98	100	94	98	98	98
Lung(L)	100	100	96	98	96	100
Lung(R)	99	98	96	96	94	100
Pelvis(L)	98	94	100	100	94	98
Pelvis(R)	99	100	98	100	94	100
Spleen	100	98	100	98	98	98

Table 2. Percentage number of detected organ walls that fall with 2 standard deviations from the mean error.

Using centroid-hit measure. Table 3 shows the centroid hit measures for the different organs of interest. The results show that the hit rate for x and y-extents are around 90% indicating that in 9 out of 10 cases, two out of the three MPR rendered images would contain the organ of interest when navigated with a single mouse click. The z-axis which corresponds to the axial plane has comparatively lower accuracy which primarily contributes towards the reduced mean accuracy. The reduced accuracy in z-axis is due to several different issues and varies from organ to organ:

- **Cropping effect**: For Lung, we often find the abdominal CT to have only bottom tenth or fifth of the lung visible in the scan. Consequently the detected bounding box often extends beyond the virtual ceiling of the image volume. This effect can be easily fixed in this case with additional heuristics that limit the detected boundary to within the image volume.
- **Poor resolution in z-axis**: Some of the scans in our database show very poor z resolution which leads to inaccuracies in the z direction. When applying the algorithm to data from more modern scanners the results are significantly improved. Here we use both old and new CT scans to ensure that the algorithm can robustly handle data stored in legacy archives.
- **Human interpretation effect**: In some cases (e.g. with pelvis), we achieve very high (over 90%) navigational success when manually testing. However, the automatically detected centroid is often affected by relatively slender high contrast bony protrusions which pushes out the centroid outside the ground truth bounding box. In other words, the 70% success rate for the pelvis is rather pessimistic when compared with the manual qualitative evaluation. In manual testing, when pelvis was present in the CT dataset, we were able to instantaneously able to reach the pelvic region all the time in dataset B.

	Heart	Kidney(L)	Kidney(R)	Liver	Lung(L)	Lung(R)	Pelvis(L)	Pelvis(R)	Spleen
All axis	94	80	74	94	60	66	70	70	84
x-axis	98	90	92	100	96	96	74	76	98
y-axis	98	88	90	100	98	98	76	76	96
z-axis	94	84	78	94	62	68	70	70	86

Table 3. **Percentage of correct organ localizations** using the centroid-hit measure.

5. CONCLUSION

Robust automated semantic annotation of DICOM images hold the potential for everyday clinical use with applications such as faster navigation, automated visualization, enhanced search and ontological association with non-image clinical data. We have demonstrated in this paper the navigational and visualization enhancement of large CT DICOM scans being facilitated by our robust, fast and accurate organ detection algorithm. The results show that the algorithm is robust and clinically accurate for the enhanced navigation use case. Our algorithm improves physician productivity by expediting navigation and visualization of large DICOM datasets with a single mouse click.

REFERENCES

[1] N. Toussaint, J. Souplet, and P. Fillard, "Medinria: Medical image navigation and research tool by inria," in *Proc. of MICCAI'07 Workshop on Interaction in medical image analysis and visualization*, (Brisbane, Australia), 2007.

[2] Y. Zhan, X.-S. Zhou, Z. Peng, and A. Krishnan, "Active scheduling of organ detection and segmentation in whole-body medical images," in *MICCAI*, 2008.

[3] S. Seifert, M. Kelm, M. Moeller, S. Mukherjee, A. Cavallaro, M. Huber, and D. Comaniciu, "Semantic annotation of medical images," *SPIE Medical Imaging* **7628**, pp. 81–8, 2010.

[4] L. Breiman, "Random forests," Tech. Rep. TR567, UC Berkeley, 1999.

[5] A. Criminisi, J. Shotton, D. Robertson, and E. Konokoglu, "Regression forests for efficient anatomy detection and localization in CT studies," in *Medical Computer Vision 2010: Recognition Techniques and Applications in Medical Imaging*, 2010.

[6] J. Gall and V. Lempitsky, "Class-specific Hough forest for object detection," in *IEEE CVPR*, (Miami), 2009.

[7] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context.," in *IJCV*, 2009.

[8] A. Montillo and H. Ling, "Age regression from faces using random forests," in *ICIP*, 2009.

[9] S. Klein, M. Staring, K. Murphy, M. Viergever, and J. P. W. Pluim, "Elastix: a toolbox for intensity based medical image registration," *IEEE Trans. Medical Imaging* **29**(1), 2010.

[10] I. Isgum, M. Staring, A. Rutten, M. Prokop, M. A. Viergever, and B. van Ginneken, "Multi-atlas-based segmentation with local decision fusionapplication to cardiac and aortic segmentation in ct scans," *IEEE Trans. Medical Imaging* **28**(7), 2009.