Endocardial 3D Ultrasound Segmentation using Autocontext Random Forests

Kevin Keraudren[1], Ozan Oktay[1], Wenzhe Shi[1],
Joseph V. Hajnal[2] and Daniel Rueckert[1]

[1] Biomedical Image Analysis Group, Imperial College London,
[2] Imaging Sciences & Biomedical Engineering Division, King's College London

Abstract. In this paper, we present the use of a generic image segmentation method, namely a succession of Random Forest classifiers in an autocontext framework, for the MICCAI 2014 Challenge on Endocardial 3D Ultrasound Segmentation (CETUS). The proposed method segments each frame independently in 90 sec, without requiring temporal information such as end-diastolic or end-systolic time points nor any registration. For better segmentation accuracy, non-local means denoising can be applied to the images at the cost of an increased run-time. The mean Dice score on the testing dataset was 84.4% without denoising and 86.4% with denoising. The originality of our approach lies in the introduction of two classes, the *myocardium* and the *mitral valve*, in addition to the *left ventricle* and the *background* classes, in order to gain contextual information for the segmentation task.

1 Introduction

The high temporal resolution of 3D ultrasound images provide cardiologists with invaluable information, enabling them to observe the heart structures and their function in real time. However, automating image analysis tasks on ultrasound images is challenging due to low signal-to-noise ratio. In this paper, we propose a method to delineate the left ventricle (LV) endocardium border in a fully automatic manner using Random Forest classifiers (RF) [1] in an autocontext framework [11].

RF have been successfully applied to various organ localization and segmentation tasks in different imaging modalities [3,6,9]. In particular, the method presented in [6] is aimed at segmenting myocardial tissues in echocardiography images. While most of the proposed image segmentation methods based on RF classify each pixel independently, we apply several RF classifiers successively, each one gaining contextual information from the classification results of their predecessors. This framework of applying successive classifiers, outlined in Fig. 1, is called *autocontext* [11]. Similarly to [5], we use geodesic distance transforms, namely the Euclidean distance weighted by the image gradient, computed between each autocontext iteration, in order to enhance contextual information.

The main contribution of this paper is to formulate the task of segmenting the LV endocardium in a way that best takes advantage of the autocontext

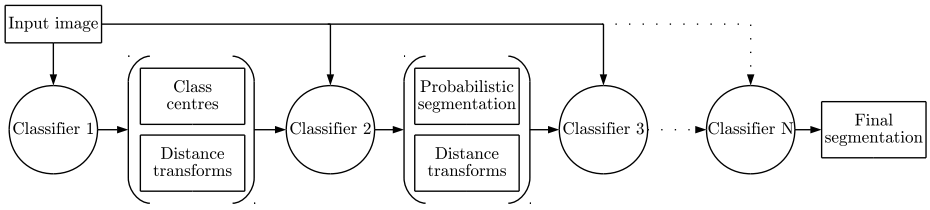


Fig. 1: Overview of the autocontext framework for segmenting the left ventricle endocardium.

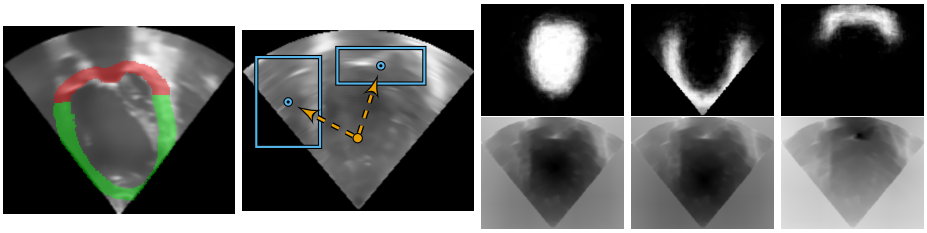
framework. Indeed, in order to increase the amount of contextual information, we formulate a four class segmentation problem by introducing classes for the *mitral valve* and *myocardium*, in addition to the *LV endocardium* and *background*. Moreover, as it may be ambiguous for a translation invariant classifier to distinguish between the four heart chambers, we limit the first iteration of autocontext to the detection of the center of each class. In the remaining of this paper, we will present the proposed method in more details, along with the results which were obtained on the first testing dataset of the CETUS challenge.

2 Method

Pre-processing: The images from all subjects are first resampled to a fixed resolution before applying non-local means denoising [2]. Denoising, which removes speckle textured image patches while preserving anatomical structures, is an optional step that slightly increases the performance of the trained classifiers.

Segmentation with autocontext Random Forests: A RF classifier is an ensemble method for machine learning that averages the results of decision trees trained on random subsets of the training dataset (*bagging*) [1]. In order to grow the decision trees, tests aiming to separate the different classes are randomly generated at every node of the trees, and the tests maximizing information gain are selected.

Similarly to [3], the tests used at the nodes of the trees are based on differences of mean intensity over displaced rectangular areas. Each test thus selects two 3D patches of random sizes at random offsets from the current pixel, and compares their mean intensity (Fig. 2.b). As these tests are invariant to intensity shifts, image intensities do not need to be standardized. In order to enable more interaction between the classes to be learnt (*spatial context*), those patches are either both selected on the original image, or on two possibly different images among the detection probability maps of each class and their corresponding geodesic distance transforms [5]. These geodesic distance transforms (GDTs) are the Euclidean distance of every pixel to the center of each class, weighted by the gradient of the image intensities (Fig. 2.c), as described in Equation 1.



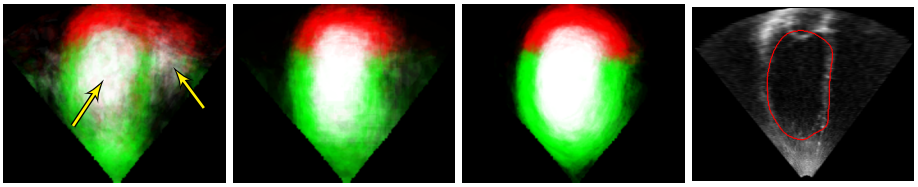
(a) (b) (c)

Fig. 2: (a) Ground truth segmentation of the myocardium (green) and the mitral valve (red) were generated using morphological operations. (b) The tests used at each tree nodes are based on the difference of mean intensity between 3D patches. (c) These patches can be taken from the input image, or from the probability maps obtained during the autocontext iterations (top row) and their corresponding GDTs (bottom row).

$$d(x,y) = \inf_{\Gamma \in \mathbf{P}_{x,y}} \int_0^{l(\Gamma)} \sqrt{1 + \gamma^2 (\nabla I(s) \cdot \Gamma'(s))^2} ds \tag{1}$$

where Γ is a path in the set of all paths $\mathbf{P}_{x,y}$ between x and y, parametrised by its arc length $s \in [0, l(\Gamma)]$, and γ is a weight between the Euclidean distance and the image gradient. GDTs, which can be efficiently computed in linear time [10], are used to increase the amount of spatial context that can be learnt by the classifier by combining prediction estimates with image information. Unlike [5] which computes GDTs using the probabilistic class regions, the class centroids are used to address the ambiguity between the four heart chambers, as highlighted in Fig. 3.a. To enable the comparison between class probabilities and GDTs, those images are rescaled to the same intensity range.

The training dataset in the CETUS challenge consists of 3D echocardiography image sequences of the beating heart, with ground truth segmentation of the left ventricle at end-diastolic (ED) and end-systolic (ES) frames. In order to train each classifier on as many images as possible (*data augmentation*), the ground truth segmentation of ED and ES frames are propagated to all other frames using non-rigid image registration. Additionally, randomly rotated versions of each frame ($\pm$ 30°along each axis) are generated to increase the training size as well as the rotation invariance of the trained classifier. Finally, we formulate the two class problem of the challenge into a four class problem: *LV endocardium*, *myocardium*, *mitral valve* and *background*. Autocontext is a framework that implicitly learns a shape model and more importantly, the spatial relation between different classes. The two additional classes are thus introduced in order to take advantage of the autocontext framework. Approximate segmentations are automatically generated for the myocardium and the mitral valve using morphological operations and fitting an ellipsoid to the ground truth segmentation of the left ventricle to obtain its main axis (Fig 2.a).



(a) (b) (c) (d)

Fig. 3: (a,b,c) Probabilistic output of the classifier for the first three iterations of autocontext for the three classes: *mitral valve* (red), *myocardium* (green) and *LV endocardium* (white). (d) Final segmentation at the 3rd iteration of autocontext (red contour). The yellow arrows in (a) point to two detected heart chambers at the first iteration of autocontext. Only the main detection is carried out through the subsequent autocontext iterations.

Although the classifier is run at every pixel when performing a segmentation, not all pixel locations are used during training in order to limit the computational cost. For the first training iteration of the classifier, the same number of pixel locations is randomly sampled within each class, forming a balanced dataset. In the subsequent iterations, pixel locations which have been correctly classified are partially masked in order to sample more pixels from the misclassified regions of the image, before randomly selecting the same number of pixel locations for each class. This is analogous to the reweighting of misclassified examples taking place in the *AdaBoost* algorithm [4].

Properties of the classifier: The proposed classifier is translation invariant and produces a probabilistic classification at every pixel. The tests performed at every node of the decision trees are designed to be invariant to intensity shifts as they compare mean intensities between patches, without setting any threshold [3]. The classifier is tolerant to some degree of rotation, depending on the amount of rotation present in the training data. By construction, it selects one connected component per class.

3 Implementation

The RF classifier is implemented in C++. Trees are grown in parallel using the TBB library. At test time, instead of parallelizing over the trees, it is more efficient to parallelize over the 2D slices of the 3D volume. The autocontext framework is written in Python. We use RF of 20 trees, with a maximal tree depth of 20. Images are resampled to $1\times1\times1$ mm^3 and the maximal patch size in the binary tests is set to 60 pixels, for a maximal offset of 30 pixels. When performing non-local means denoising, the patch size is set to $3\times3\times3$ and the weights of the neighboring patches are computed within an image window of size $7\times7\times7$ centred on the target patch. Training the classifiers takes approximately a day on a 32 cores, 256 GB RAM computer, while testing only takes 90 sec for 4

Table 1: Segmentation results on the testing set (Patients 16 to 30): mean absolute distance (MAD), Hausdorff distance (HD) and modified Dice score (DS).

	Original images			Speckle reduced images		
Training size	MAD (mm)	HD (mm)	DS (%)	MAD (mm)	HD (mm)	DS (%)
100 frames	2.70±0.84	9.58±3.36	16.1±5.5	2.75±1.01	9.58±3.13	15.7±5.5
300 frames	2.60±0.80	9.64±3.28	15.6±5.4	2.32±0.69	9.19±3.33	14.3±5.2
600 frames	2.57±0.68	9.20±3.08	15.7±5.6	2.31±0.74	8.66±2.79	13.6±4.2

autocontext iterations. Non-local means denoising for a single frame takes about 1 min on a quad-core machine. Due to the time required to train the classifier, no cross-validation was performed on the training data.

As a post-processing step, the probabilistic segmentations are upsampled to $0.5 \times 0.5 \times 0.5$ mm^3, a Gaussian filter of standard deviation 0.5 mm is applied, and the iso-surface for probability 0.5 is extracted using the marching cubes algorithm [7].

4 Results and discussion

The results obtained on the first testing dataset of the CETUS challenge are presented in Table 1 for different sizes of training dataset. The best segmentation scores were obtained on the denoised data, for a training size of 600 frames and 4 iterations of autocontext, with a mean Dice score of 86.4%. The best results obtained are detailed in Table 2.a, with distinct scores for ED and ES frames. Among all the parameters of the model, the two most important were the size of the training dataset and the choice of additional classes that can provide spatial context when learning to segment the left ventricle, such as the mitral valve and the myocardium (Fig. 2.a). Indeed, in the experiments performed, introducing a class for the mitral valve was found necessary to enable the classifier to position the boundary between the left ventricle and the left atrium.

Two clinical parameters, stroke volume (SV) and ejection fraction (EF), were evaluated for each patient. They are defined as follows:

$$SV = EDV - ESV \tag{2}$$

$$EF = \frac{EDV - ESV}{EDV} \times 100 \tag{3}$$

where EDV is the end-diastolic volume and ESV denotes the end-systolic volume. These parameters were compared against their reference values, and the correlation coefficients, bias and limits of agreement are reported in Table 2.b. A strong correlation can be observed between EDV, EDS and their reference values (respectively 0.917 and 0.979). The lack of correlation between the measured stroke volume and its reference value (correlation coefficient 0.045) may be

explained by the fact that ED and ES frames are segmented independently, using the same trained classifier. A model which would take time information into account and segment each frame in the context of the preceding and successive frames might provide a more accurate estimate of the stroke volume and ejection fraction. Besides, as can be seen in Table 2.a, the proposed model does not perform as well on ES frames as on ED frames. This observation could motivate training distinct ED and ES classifiers.

Table 2: Segmentation results on the testing set (Patients 16 to 30) for end-diastolic and end-systolic frames, on speckle reduced images, using 600 frames for training: (a) mean absolute distance (MAD), Hausdorff distance (HD) and modified Dice score (DS); (b) correlation coefficient (CC), bias and limit of agreement (LOA) for the ejection fraction and stroke volume indices.

(a)

	MAD (mm)	HD (mm)	DS (%)
End-diastolic	2.28±0.92	8.29±2.37	12.1±4.3
End-systolic	2.33±0.50	9.03±3.12	15.0±3.6

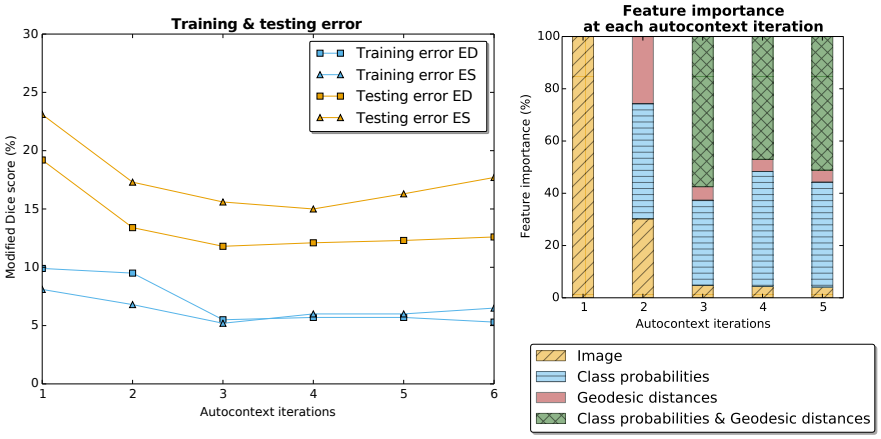
(b)

	CC	Bias	LOA ($\mu \pm 1.96\sigma$)
End-diastolic volume (mL)	0.917	6.61	-30.85 to 44.07
End-systolic volume (mL)	0.979	-7.85	-32.21 to 16.51
Stroke volume (mL)	0.045	14.43	-26.80 to 55.65
Ejection fraction (%)	0.780	8.49	-12.58 to 29.57

The importance of autocontext is highlighted in Fig. 3.a. Indeed, the first iteration of autocontext, which classifies each pixel independently, might detect more than one heart chamber due to the translation invariance of the classifier. This first iteration is thus only used to define the center of each class, allowing the subsequent iterations to focus on the correct region of the image. In order to demonstrate the feasibility of segmenting the whole temporal sequence in addition to the ED and ES frames using the proposed method, a video is available online[1].

Feature importance in Random Forests is a mean to measure which tests are most capable of separating the different classes. The most important tests are selected early in the tree construction, and repetitively across the forest. Figure 4 indicates which tests play a more important role at the different iterations of autocontext. These tests can be summarized into four categories: the comparison of patches within the original image, patches across class probabilities, patches across GDTs, and patches between both class probabilities and GDTs. It can be noted that during the first iteration of autocontext, all tests are made on the original image and that during the second iteration, no tests are made

[1] `http://www.doc.ic.ac.uk/~kpk09/MICCAI2014_CETUS.mp4`



(a) (b)

Fig. 4: (a) Modified Dice score on the training and testing datasets, using 600 frames, for end-diastolic (ED) and end-systolic frames (ES). (b) Feature ranking at each autocontext iteration: the tests performed in the tree nodes compare the mean intensity of two patches which can originate from the original image, or from different class probabilities, or different geodesic distance transforms, or a class probability and a geodesic distance transform.

between the GDTs and the class probabilities as the latter only contains the class centroids. In the subsequent iterations, only a minority of tests take place on the original image. This is a form of overfitting as the algorithm no longer uses the original image and instead recreates the shapes it learned during training. In the example video, overfitting can be observed in the last frames where the segmentation of the LV overlaps with the endocardium, despite its boundary being clearly defined in the original image. Overfitting can be observed as well in Figure 4.a as the performance of the classifier on the test dataset decreases after the 4^{th} iteration while its performance on the training dataset is almost constant.

The anatomy of the left ventricle is more complex than the simplified model used to train the classifier. A more realistic and detailed ground truth, for instance taking into account both the mitral valve and the aortic valve, could potentially improve the accuracy of the classifier.

5 Conclusion

We presented a generic image segmentation method, autocontext Random Forests, applied to the segmentation of the left ventricle endocardium in 3D echocardiography images. The only part of the method which is task specific is the choice of the different classes to segment, as these classes must enable the classifier to learn spatial context. This method can be applied to any time frame of an

echocardiography sequence in a reasonable time. Future work will investigate different sets of tests for the decision trees, such as the use of local binary pattern (LBP) features [8].

References

1. Breiman, L.: Random Forests. Machine Learning 45(1), 5–32 (2001)
2. Coupé, P., Hellier, P., Kervrann, C., Barillot, C.: Nonlocal Means-based Speckle Filtering for Ultrasound Images. IEEE Transactions on Image Processing 18(10), 2221–29 (2009)
3. Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E.: Regression Forests for Efficient Anatomy Detection and Localization in CT Studies. Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging pp. 106–117 (2011)
4. Freund, Y., Schapire, R.E.: A Desicion-theoretic Generalization of On-line Learning and an Application to Boosting. In: Computational Learning Theory. pp. 23–37. Springer (1995)
5. Kontschieder, P., Kohli, P., Shotton, J., Criminisi, A.: GeoF: Geodesic Forests for Learning Coupled Predictors. In: Computer Vision and Pattern Recognition (CVPR). pp. 65–72. IEEE (2013)
6. Lempitsky, V., Verhoek, M., Noble, J.A., Blake, A.: Random Forest Classification for Automatic Delineation of Myocardium in Real-time 3D Echocardiography. In: Functional Imaging and Modeling of the Heart, pp. 447–456. Springer (2009)
7. Lorensen, W.E., Cline, H.E.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In: Siggraph Computer Graphics. vol. 21, pp. 163–169. ACM (1987)
8. Ojala, T., Pietikäinen, M., Harwood, D.: A Comparative Study of Texture Measures with Classification based on Featured Distributions. Pattern Recognition 29(1), 51–59 (1996)
9. Pauly, O., Glocker, B., Criminisi, A., Mateus, D., Möller, A., Nekolla, S., Navab, N.: Fast Multiple Organ Detection and Localization in Whole-body MR Dixon Sequences. In: MICCAI. pp. 239–247. Springer (2011)
10. Toivanen, P.J.: New Geodesic Distance Transforms for Gray-scale Images. Pattern Recognition Letters 17(5), 437 – 450 (1996)
11. Tu, Z.: Auto-Context and its Application to High-level Vision Tasks. In: Computer Vision and Pattern Recognition (CVPR). pp. 1–8. IEEE (2008)