# Iterative Segmentation from Limited Training Data: Applications to Congenital Heart Disease

**Danielle F. Pace**[1], **Adrian V. Dalca**[1,2,3], **Tom Brosch**[4], **Tal Geva**[5,6], **Andrew J. Powell**[5,6], **Jürgen Weese**[4], **Mehdi H. Moghari**[5,6], and **Polina Golland**[1]

[1]Computer Science and Artificial Intelligence Lab, MIT, Cambridge, USA

[2]Martinos Center for Biomedical Imaging, Massachusetts General Hospital, HMS, Boston, USA

[3]School of Electrical and Computer Engineering, Cornell University, Ithaca, USA

[4]Philips Research Laboratories, Hamburg, Germany

[5]Department of Cardiology, Boston Children's Hospital, Boston, USA

[6]Department of Pediatrics, Harvard Medical School, Boston, USA

## Abstract

We propose a new iterative segmentation model which can be accurately learned from a small dataset. A common approach is to train a model to directly segment an image, requiring a large collection of manually annotated images to capture the anatomical variability in a cohort. In contrast, we develop a segmentation model that recursively evolves a segmentation in several steps, and implement it as a recurrent neural network. We learn model parameters by optimizing the intermediate steps of the evolution in addition to the final segmentation. To this end, we train our segmentation propagation model by presenting incomplete and/or inaccurate input segmentations paired with a recommended next step. Our work aims to alleviate challenges in segmenting heart structures from cardiac MRI for patients with congenital heart disease (CHD), which encompasses a range of morphological deformations and topological changes. We demonstrate the advantages of this approach on a dataset of 20 images from CHD patients, learning a model that accurately segments individual heart chambers and great vessels. Compared to direct segmentation, the iterative method yields more accurate segmentation for patients with the most severe CHD malformations.

## 1 Introduction

We aim to provide whole heart segmentation in cardiac MRI for patients with congenital heart disease (CHD). This involves delineating the heart chambers and great vessels [1], and promises to enable patient-specific heart models for surgical planning in CHD [2]. CHD encompasses a vast range of cardiac malformations and topological changes. Defects can include holes in the heart walls (septal defects), great vessels connected to the wrong

---

dfpace@mit.edu.

chamber (e.g., double outlet right ventricle; DORV), dextrocardia (left-right flip), duplication of a great vessel, a single ventricle, and/or prior surgeries creating additional atypical connections. In MRI, different chambers and great vessels locally appear very similar to each other, and there is little or no contrast at the valves and thin walls separating neighboring structures. Finally, labeled training data is very limited. This precludes modeling each CHD subtype separately in an attempt to reduce variability. Moreover, patients with unique combinations of defects and prior surgeries defy categorization. Beyond our application, limited training data is to be expected for new applications of medical imaging not yet in widespread clinical practice. This necessitates development of methods that generalize well from small, imbalanced datasets, possibly also incorporating user interaction.

State-of-the-art methods use a convolutional neural network (CNN) to directly outline all chambers and vessels in one step [3,4]. However, CNNs for CHD have largely been limited to segmenting the blood pool and myocardium [5,6]. Direct co-segmentation of all major cardiac structures works well when applied to adult-onset heart disease, which induces much less severe shape changes compared to CHD. However, it fails completely on held-out subjects with severe CHD malformations after training with our small dataset of CHD patients.

We develop an iterative segmentation approach that evolves a segmentation over several steps in a prescribed way and automatically estimates when to stop, beginning from a single seed for each structure placed by the user. An iterative method can operate more locally, better maintain each structure's connectivity, and propagate information from distant landmarks, similar to traditional snakes, level sets and particle filters [7]. We employ a recurrent neural network (RNN) [8], which uses context to grow the segmentation appropriately even in areas of low contrast. Deep learning research has indeed focused on segmenting a single image iteratively. Examples include recursive refinement of the entire segmentation map [9,10], sequential completion of different instances, regions or fields of view [11–13], slice-by-slice analysis [14] and networks modeling level set evolution [15]. These methods condition on a previous partial solution to make progress towards the final output. This simplified task may enable training from smaller datasets.

We train the model by minimizing a loss over a training dataset of example segmentation *trajectories*. Maximizing the likelihood of observed sequences is known as teacher forcing [8,16]. For example, we may require vessel segmentation to proceed at a constant rate along the vessel centerline, or a heart chamber segmentation to dilate outwards. Even if the stopping prediction is incorrect, since the segmentation evolution follows a prescribed pattern it is likely that one of the intermediate segmentations will be accurate. In contrast, using the final segmentation alone could lead to unpredictable growth patterns. Teacher forcing also leads to a simplified optimization over decoupled time steps, avoiding back-propagation through time.

We focus on segmenting the aorta (a representative great vessel) and the left ventricle (a representative cardiac chamber). We validate our iterative segmentation approach using a

dataset of 20 CHD patients, and compare it to direct segmentation methods which we have developed for this problem.

## 2 Iterative Segmentation Model

Given an input image $\mathbf{x}$ defined on the domain $\Omega$, we seek a segmentation label map $\mathbf{y}$ that assigns one of $L$ anatomical labels to each voxel in $\mathbf{x}$.

**Generative Model:**

We model the segmentation $\mathbf{y}$ as the endpoint of a sequence of segmentations $\mathbf{y}_0, \ldots, \mathbf{y}_T$, where $\mathbf{y}_t : \Omega \rightarrow \{1, \ldots, L\}$ for time steps $t = 0, \ldots, T$. The intermediate segmentations $\mathbf{y}_t$ capture a growing part of the anatomy of interest. In practice, the initial segmentation map $\mathbf{y}_0$ is created by centering a small sphere around an initial seed point placed by the user.

The number of iterations required to achieve an accurate segmentation depends on the shape and size of the object being segmented. To capture this, we introduce a sequence of indicator variables $s_0, \ldots, s_T$, where $s_t \in \{0, 1\}$ specifies whether the segmentation is completed at time step $t$. If $s_t = 1$, then $\mathbf{y}_t$ is deemed the final segmentation and we set $\mathbf{y}_i = \mathbf{y}_{i-1}$ and $s_i = 1$ for all $i > t$.

Given an image and an initial segmentation, the inference task is to compute $p(\mathbf{y}_T, s_T | \mathbf{x}, \mathbf{y}_0, s_0 = 0)$. We assume that the segmentations $\{\mathbf{y}_t\}$ and stopping indicators $\{s_t\}$ follow a first order Markov chain given the input image:

$$p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_0, \ldots, \mathbf{y}_{t-1}, s_0, \ldots, s_{t-1}) = p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1}), \quad (1)$$

$$p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_0, s_0) = \sum_{\mathbf{y}_{t-1}} \sum_{s_{t-1}} p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1}) \cdot p(\mathbf{y}_{t-1}, s_{t-1} \mid \mathbf{x}, \mathbf{y}_0, s_0). \quad (2)$$

**Transition Probability Model:**

We must define the transition probability $p(\mathbf{y}_t, s_t | \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1})$ to complete the recursion in Eq. (2). There are two possible cases: $s_{t-1} = 1$ and $s_{t-1} = 0$. Based on the definition of $s_{t-1}$, we obtain

$$p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 1) = \mathbb{1}(\mathbf{y}_t = \mathbf{y}_{t-1}) \cdot \mathbb{1}(s_t = 1), \quad (3)$$

where $\mathbb{1}(\cdot)$ denotes the indicator function. To compute $p(\mathbf{y}_t, s_t | \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$, we introduce a latent representation $\mathbf{h}_t = h(\mathbf{x}, \mathbf{y}_{t-1})$ that jointly captures all of the necessary information from image $\mathbf{x}$ and previous segmentation $\mathbf{y}_{t-1}$. Intuitively, predicting whether the segmentation $\mathbf{y}_t$ is complete given $\mathbf{x}$ can be performed by examining whether $\mathbf{y}_{t-1}$ is

"almost" complete. Therefore, the segmentation $\mathbf{y}_t$ and stopping indicator $s_t$ are conditionally independent given $\mathbf{h}_t$:

$$p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0) = p(\mathbf{y}_t, s_t \mid \mathbf{h}_t) = p(\mathbf{y}_t \mid \mathbf{h}_t) \cdot p(s_t \mid \mathbf{h}_t). \quad (4)$$

We model the function $h(\mathbf{x}, \mathbf{y}_{t-1})$ and distributions $p(\mathbf{y}_t | \mathbf{h}_t)$ and $p(s_t | \mathbf{h}_t)$ as stationary; they do not depend on the time step $t$.

### Learning:

We learn a representation of $p(\mathbf{y}_t, s_t | \mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$ given a training dataset of example desired trajectories of segmentations. Specifically, we consider a training dataset $\mathscr{D}$ of $N$ images $\{\mathbf{x}^i\}_{i=1}^N$, each of which has a corresponding sequence of segmentations $\mathbf{y}_0^i, ..., \mathbf{y}_{T_i}^i$ and of stopping indicators $s_0^i, ..., s_{T_i}^i$, where $s_0^i = ... = s_{T_{i-1}}^i$ and $s_{T_i}^i = 1$. The parameter values to be determined are $\boldsymbol{\theta} = \{\boldsymbol{\theta}_h, \boldsymbol{\theta}_y, \boldsymbol{\theta}_s\}$ corresponding to $h(\mathbf{x}, \mathbf{y}_{t-1}; \boldsymbol{\theta}_h)$, $p(\mathbf{y}_t | \mathbf{h}_t; \boldsymbol{\theta}_y)$, and $p(s_t | \mathbf{h}_t; \boldsymbol{\theta}_s)$, respectively. We seek the parameter values that minimize the expected negative log-likelihood of the output segmentation and stopping indicator sequences given the image and initial conditions, i.e., $\boldsymbol{\theta}^* = \mathrm{argmin}_\theta \mathscr{L}(\boldsymbol{\theta})$,

$$\mathscr{L}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x}, \mathbf{y}_0, ..., \mathbf{y}_T, s_0, ..., s_T \sim \mathscr{D}} \left[ -\log p(\mathbf{y}_1, ..., \mathbf{y}_T, s_1, ..., s_T \mid \mathbf{x}, \mathbf{y}_0, s_0; \boldsymbol{\theta}) \right] \quad (5)$$

$$= -\mathbb{E} \left[ \sum_{t=1}^T \log p(\mathbf{y}_t \mid h(\mathbf{x}, \mathbf{y}_{t-1}; \boldsymbol{\theta}_h); \boldsymbol{\theta}_y) + \log p(s_t \mid h(\mathbf{x}, \mathbf{y}_{t-1}; \boldsymbol{\theta}_h); \boldsymbol{\theta}_s) \right].$$

Note that teacher forcing has lead to decoupled time steps. The first and second terms in the likelihood above penalize differences for the segmentations and the stopping indicators, respectively, between the predicted probabilities and the ground truth. In practice, we perform class rebalancing for both terms, and further supplement the segmentation loss by more strongly weighting pixels on the boundaries of the ground truth segmentation.

### Inference:

Computing $p(\mathbf{y}_T, s_T | \mathbf{x}, \mathbf{y}_0, s_0 = 0)$ via the recursion in Eq. (2) is intractable due to the summation over all possible segmentations $\mathbf{y}_{t-1}$. To approximate, we follow a widely accepted practice of using the most likely segmentation $\mathbf{y}_{t-1}^*$ and stopping indicator $s_{t-1}^*$ as input to the subsequent computation:

$$p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_0, s_0 = 0; \boldsymbol{\theta}) \approx p(\mathbf{y}_t, s_t \mid \mathbf{x}, \mathbf{y}_{t-1}^*, s_{t-1}^*; \boldsymbol{\theta}), \quad (6)$$

$$\text{where } \mathbf{y}_{t-1}^*, s_{t-1}^* = \underset{\mathbf{y}_{t-1}, s_{t-1}}{\mathrm{argmax}} \; p(\mathbf{y}_{t-1}, s_{t-1} \mid \mathbf{x}, \mathbf{y}_0, s_0 = 0; \boldsymbol{\theta}).$$

The segmentation is fully automatic given the initial seed. If the stopping indicator is predicted incorrectly, a user can manually override it by asking for more iterations or by choosing a segmentation from a previous step.

**RNN:**

We implement our iterative segmentation model as an RNN (Fig. 1), which is formed by connecting identical copies of an augmented 3D U-net [17] trained to estimate $p(\mathbf{y}_t, s_t|\mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$. Thus, parameters are shared both spatially and temporally. At each step, the U-net inputs the image and the most likely segmentation from the previous step. This respects the Markov property in Eq. (1), unlike if any hidden layers were connected between successive steps. If the stopping indicator $s_t^* = 1$, the segmentation propagation halts.

Our augmented U-net modeling $p(\mathbf{y}_t, s_t|\mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$ has $L + 1$ input channels, containing the input image and a binary map for each of the $L$ labels in the segmentation $\mathbf{y}_{t-1}$ (including the background). There are two outputs: the probability map for the segmentation $\mathbf{y}_t$ (at each voxel, representing the parameters of the categorical distribution over $L$ labels), and the Bernoulli stopping parameter $p(s_t = 1|\mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$. Jointly predicting the segmentation and stopping indicator enables a smaller model compared to two separate networks.

The original U-net for image segmentation produces a final set of $C$ learned feature maps, which undergo $C \cdot L$ $1 \times 1 \times 1$ convolutions and a softmax activation to give the output segmentation probabilities. We use these $C$ learned feature maps as the latent joint representation $\mathbf{h}_t = h(\mathbf{x}, \mathbf{y}_{t-1}; \boldsymbol{\theta}_h)$. The U-net parameters can therefore be split into two sets. The parameters for the final $1 \times 1 \times 1$ convolutions are $\boldsymbol{\theta}_y$ of $p(\mathbf{y}_t|\mathbf{h}_t; \boldsymbol{\theta}_y)$, and the remainder are $\boldsymbol{\theta}_h$ of $h(\mathbf{x}, \mathbf{y}_{t-1}; \boldsymbol{\theta}_h)$. The probability $p(s_t = 1|\mathbf{h}_t; \boldsymbol{\theta}_s)$ is computed by applying $C$ additional $3 \times 3 \times 3$ convolutions with parameters $\boldsymbol{\theta}_s$ to the feature maps in $\mathbf{h}_t$, followed by a global average and sigmoid activation to yield a scalar in $\{0, 1\}$.

**Generating Segmentation Trajectories:**

Our training dataset of images and segmentation trajectories is derived from a collection of paired images and complete segmentations. Several acceptable trajectories exist for each pair, e.g., starting from different initial seeds. To this end, at the beginning of each epoch a random tuple $(\mathbf{y}_{t-1}, \mathbf{y}_t, s_t)$ is generated for each image. These tuples all follow the same principle that we want the network to learn.

As a concrete example, the trajectories used in our experiments are as follows. For the *aorta*, the segmentation grows from the seed along the vessel centerline, by a random distance to form $\mathbf{y}_{t-1}$ and an additional 10 pixels for $\mathbf{y}_t$. The seed is placed in the descending aorta, and the endpoint is at the valve where the aorta connects to a left or right ventricle. This seed could be automatically detected in the future, and the lack of contrast at the valve provides a challenging test case for our automatic stopping. For the *left ventricle*, we randomly place the seed in the center region of the chamber, and perform a random number of dilations to form $\mathbf{y}_{t-1}$, and 3 more dilations to form $\mathbf{y}_t$.

**Data Augmentation:**

Data augmentation is essential to prevent overfitting on a small training dataset. We mimic the diversity of heart shapes and sizes, global intensity changes caused by inhomogeneity artifacts, and noise induced by elevated heart rates or arrhythmias. We apply random rigid and nonrigid transformations, random constant intensity shifts, and random additive Gaussian noise. We also investigate including random left-right (L-R) and anterior-posterior (A-P) flips, to better handle dextrocardia or other cardiac malpositions, since in these cases the left ventricle may lie on the right side of the body.

If the augmented U-net for $p(\mathbf{y}_t, s_t|\mathbf{x}, \mathbf{y}_{t-1}, s_{t-1} = 0)$ is trained solely using error-free segmentations $\mathbf{y}_{t-1}$, then it may not operate well on its own imperfect intermediate results at test time. We increase robustness by performing additional data augmentation on the input segmentations $\mathbf{y}_{t-1}$. We corrupt these segmentations by applying random nonrigid deformations, and by inserting random blob-like structures that vary in number, location and size and are attached to the segmentation foreground or free-floating. Since the target segmentation $\mathbf{y}_t$ remains unchanged, the model learns to correct mistakes in its input.

## 3 Experimental Validation

We evaluate our iterative segmentation and tailored direct segmentation methods, focusing on segmenting the aorta and left ventricle (LV) of CHD patients.

**Data:**

We use the HVSMR dataset of 20 MRI scans from patients with a variety of congenital heart defects [18]. Each high-resolution ($\approx$0.9mm$^3$) 3D image was acquired on a 1.5T scanner (Philips Achieva), without contrast agent and using a free-breathing SSFP sequence with ECG and respiratory navigator gating. The HVSMR dataset includes blood pool and myocardium segmentations only. A trained rater manually separated all of the heart chambers and great vessels. The 20 images were categorized after visually assessing any gross morphological malformations: 4/20 severe (prior major reconstructive surgery, single ventricle, dextrocardia), 5/20 moderate (DORV, VSD, abnormal chamber shapes), and 11/20 mild (ASD, stenosis, etc.). The dataset was randomly split into 4 folds for cross-validation (15 training, 5 testing), with an equal number of mild, moderate and severe cases in each. Input images were resized to $\approx$128 $\times$ 180 $\times$ 144.

**Experiments:**

In our tests, binary segmentation of each structure outperformed co-segmenting all of the heart chambers and vessels. We trained several models aimed at segmenting the aorta and left ventricle of CHD patients. **DIR** uses a single U-net to perform direct binary segmentation. **DIR-DIST** includes the Euclidean distance to the initial seed as an additional input channel. **ITER (stop)** is iterative segmentation using our RNN with automatic stopping, and **ITER (max)** simulates a user by choosing the segmentation with the best Dice coefficient after 30 iterations of our RNN. Finally, **ITER-SEG-ABL** is an ablation study with no data augmentation on the input segmentations. We tuned the architectural

parameters for each experiment separately, nevertheless resulting in similar networks. All U-nets had 3 levels, 24 feature maps at the first level, and ≈870,000 parameters. The best network for direct segmentation of the aorta used $2 \times 2 \times 2$ max pooling (receptive field = $40^3$), while all others used $3 \times 3 \times 3$ max pooling (receptive field = $68^3$). For training, optimization using adadelta ran for 2000 epochs with a batch size of 1. For iterative segmentation, the argmax in Eq. (6) is computed per voxel, by assuming that the segmentation of each voxel is conditionally independent of all other voxels given $\mathbf{h}_t$. Segmentations were post-processed to keep only the largest island or the island containing the initial seed, for experiments in which this improves overall accuracy. Aorta segmentations were not penalized for descending aortas longer than in the gold-standard.

**Results:**

Figures 2 and 3 report the results. There was no notable difference in accuracy between the mild and moderate groups. DIR-DIST was the best direct segmentation method, demonstrating the advantage of leveraging user interaction. For all methods, incorporating L-R and A-P flips in the data augmentation improved performance for severe subjects. Iterative segmentation stopped automatically after $18 \pm 3$ steps for both the aorta and the LV, requiring ≈15 s. The potential benefits of our iterative segmentation approach are demonstrated by the performance of ITER (max), which shows improvement for all of the severe cases while maintaining accuracy for the others. The stopping prediction is not perfect at test time: the number of iterations separating the automatic stopping point from the best segmentation in a sequence was $0.8 \pm 1.0$ iterations for the aorta and $3.0 \pm 2.5$ iterations for the LV. The sole aorta containing a stent was poorly segmented by all methods (Fig. 3e). The stent caused a strong inhomogeneity artifact that the iterative segmentation could not grow past, and the stopping criterion was never triggered.

## 4 Conclusions

We presented an iterative segmentation model and its RNN implementation. We showed that for whole heart segmentation, the iterative approach was more robust to the cardiac malformations of severe CHD. Future work will investigate the potential general applicability of iterative segmentation when one is restricted to a small training dataset despite wide anatomical variability.

## Acknowledgements.

## References

1. Zhuang X: Challenges and methodologies of fully automatic whole heart segmentation: a review. J. Healthc. Eng 4(3), 371–408 (2013) [PubMed: 23965595]

2. Pace DF, Dalca AV, Geva T, Powell AJ, Moghari MH, Golland P: Interactive whole-heart segmentation in congenital heart disease In: Navab N, Hornegger J, Wells WM, Frangi AF (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 80–88. Springer, Cham (2015). 10.1007/978-3-319-24574-4_10

3. Payer C, Štern D, Bischof H, Urschler M: Multi-label whole heart segmentation using CNNs and anatomical label configurations In: Pop M, et al. (eds.) STACOM 2017. LNCS, vol. 10663, pp. 190–198. Springer, Cham (2018). 10.1007/978-3-319-75541-0_20

4. Wang C, Smedby Ö: Automatic whole heart segmentation using deep learning and shape context In: Pop M, et al. (eds.) STACOM 2017. LNCS, vol. 10663, pp. 242–249. Springer, Cham (2018). 10.1007/978-3-319-75541-0_26

5. Wolterink JM, Leiner T, Viergever MA, Išgum I: Dilated convolutional neural networks for cardiovascular MR segmentation in congenital heart disease In: Zuluaga MA, Bhatia K, Kainz B, Moghari MH, Pace DF (eds.) RAMBO/HVSMR −2016. LNCS, vol. 10129, pp. 95–102. Springer, Cham (2017). 10.1007/978-3-319-52280-7_9

6. Yu L, Yang X, Qin J, Heng P-A: 3D FractalNet: dense volumetric segmentation for cardiovascular MRI volumes In: Zuluaga MA, Bhatia K, Kainz B, Moghari MH, Pace DF (eds.) RAMBO/HVSMR −2016. LNCS, vol. 10129, pp. 103–110. Springer, Cham (2017). 10.1007/978-3-319-52280-7_10

7. Sonka M, Hlavac V, Boyle R: Image Processing, Analysis and Machine Vision. Thompson, Toronto (2008)

8. Goodfellow I, Bengio Y, Courville A: Deep Learning. MIT Press, Cambridge (2016)

9. Pinheiro P, Collobert R: Recurrent convolutional neural networks for scene labeling. In: ICML pp. I-82–I-90 (2014)

10. Zhou Y, Xie L, Shen W, Wang Y, Fishman EK, Yuille AL: A fixed-point model for pancreas segmentation in abdominal CT scans In: Descoteaux M, Maier-Hein L, Franz A, Jannin P, Collins DL, Duchesne S (eds.) MICCAI 2017. LNCS, vol. 10433, pp. 693–701. Springer, Cham (2017). 10.1007/978-3-319-66182-7_79

11. Ren M, Zemel R: End-to-end instance segmentation with recurrent attention. In: CVPR, pp. 6656–6664 (2017)

12. Banica D, Sminchisescu C: Second-order constrained parametric proposals and sequential search-based structured prediction for semantic segmentation in RGB-D images. In: CVPR, pp. 3517–3526 (2015)

13. Januszewski M, et al.: High-precision automated reconstruction of neurons with flood-filling networks. Nat Methods, Preprint (2018)

14. Zheng Q, Delingette H, Duchateau N, Ayache N: 3D consistent and robust segmentation of cardiac images by deep learning with spatial propagation. IEEE Trans. Med. Imaging, Preprint (2018)

15. Chakravarty A, Sivaswamy J: RACE-net: a recurrent neural network for biomedical image segmentation. IEEE J. Biomed. Health Inform, Preprint (2018)

16. Williams R, Zipser D: A learning algorithm for continually running fully recurrent neural networks. Neural Comput. 1(2), 270–280 (1989)

17. Ronneberger O, Fischer P, Brox T: U-Net: convolutional networks for biomedical image segmentation In: Navab N, Hornegger J, Wells WM, Frangi AF (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). 10.1007/978-3-319-24574-4_28

18. HVSMR Challenge, MICCAI (2016). axial-cropped, http://segchd.csail.mit.edu
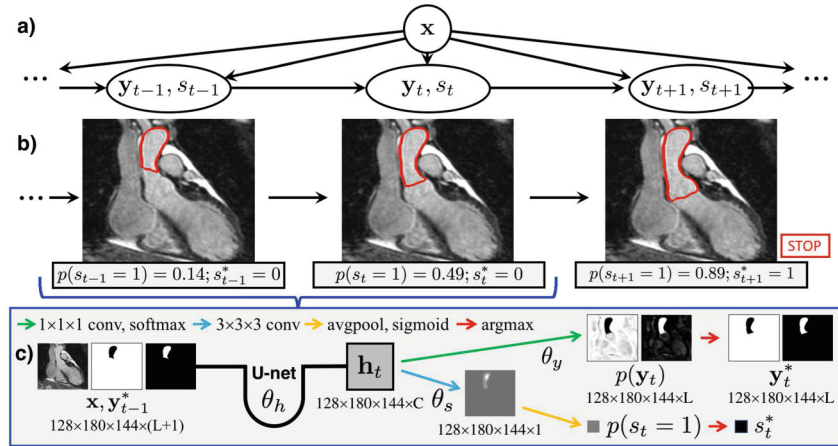
**Fig. 1.**
Iterative segmentation as an RNN. (a) Generative model. (b) The RNN uses the same augmented U-net at each step to predict the next segmentation and stopping indicator. (c) Architecture details (conditioning dropped for clarity).
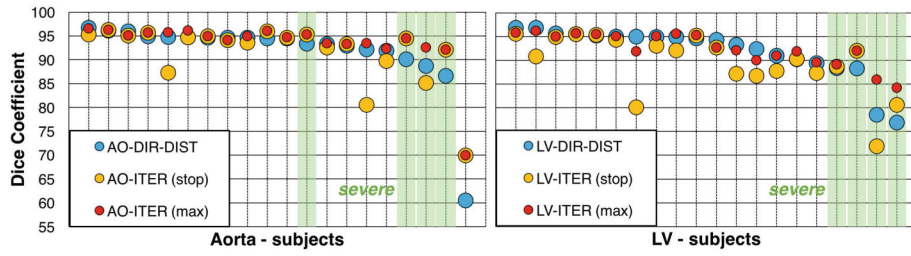
**Fig. 2.**

Aorta (AO) and LV segmentation validation. DIR-DIST is the best direct segmentation method, but iterative segmentation generalizes better to severe subjects. Top: Dice coefficients for all methods. Bottom: Results for all 20 subjects, sorted by DIR-DIST score and with severe subjects highlighted in green. (Color figure online)
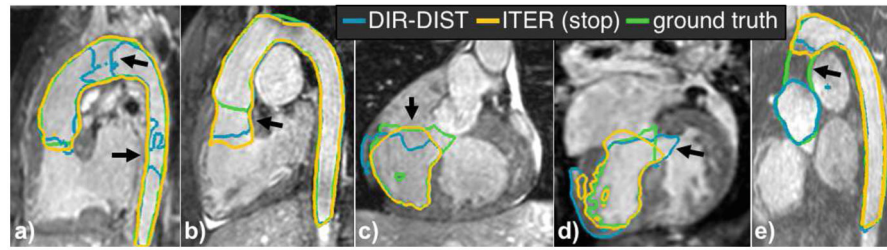
**Fig. 3.**

Representative aorta and LV segmentations in held-out subjects with severe CHD. Arrows illustrate both the benefits and failure cases of iterative segmentation with automatic stopping, where it (a) successfully segments a difficult case, (b) stops too late, (c) correctly stops near a valve, (d) avoids growing through a septal defect, (e) cannot grow through a dark region caused by a stent.

| Method | AO mild/mod. | AO severe | LV mild/mod. | LV severe |
|---|---|---|---|---|
| DIR | 92.5±6.5 | 81.2±16.3 | 94.1±3.5 | 68.6±25.5 |
| DIR-DIST | 92.3±8.6 | 89.7±2.9 | 94.1±2.2 | 83.0±6.2 |
| ITER (stop) | 91.5±7.0 | 91.8±4.6 | 91.2±4.4 | 83.3±9.0 |
| ITER (max) | 93.3±6.3 | 93.6±1.5 | 93.7±2.3 | 87.8±3.5 |
| ITER-SEG-ABL (stop) | 65.9±24.1 | 45.0±33.4 | 62.2±24.9 | 49.2±31.3 |
| ITER-SEG-ABL (max) | 66.3±24.4 | 45.8±37.4 | 64.4±22.4 | 52.7±25.1 |