

Elastic Boundary Projection for 3D Medical Imaging Segmentation

Tianwei Ni¹, Lingxi Xie², Huangjie Zheng³, Elliot K. Fishman⁴, Alan L. Yuille²

¹Peking University ²The Johns Hopkins University

³Shanghai Jiao Tong University ⁴The Johns Hopkins Medical Institute

{twni2016, 198808xc, alan.l.yuille}@gmail.com

zhj865265@sjtu.edu.cn efishman@jhmi.edu

Presented by: Xi Fang
07/24/2019

Motivation & Contribution

	2D-Net [21][29]	3D-Net [20][31]	AH-Net [18]	EBP (ours)
Pure 2D network?	✓			✓
Pure 3D network?		✓		
Working on 3D data?		✓	✓	✓
3D data not cropped?				✓
3D data not rescaled?				✓
Can be pre-trained?	✓		✓	✓

Table 1. A comparison between EBP and previous approaches in network dimensionality, data dimensionality, and the ways of pre-processing data and network weights. Due to space limit, we do not cite all related work here – see Section 2 for details.

Methods

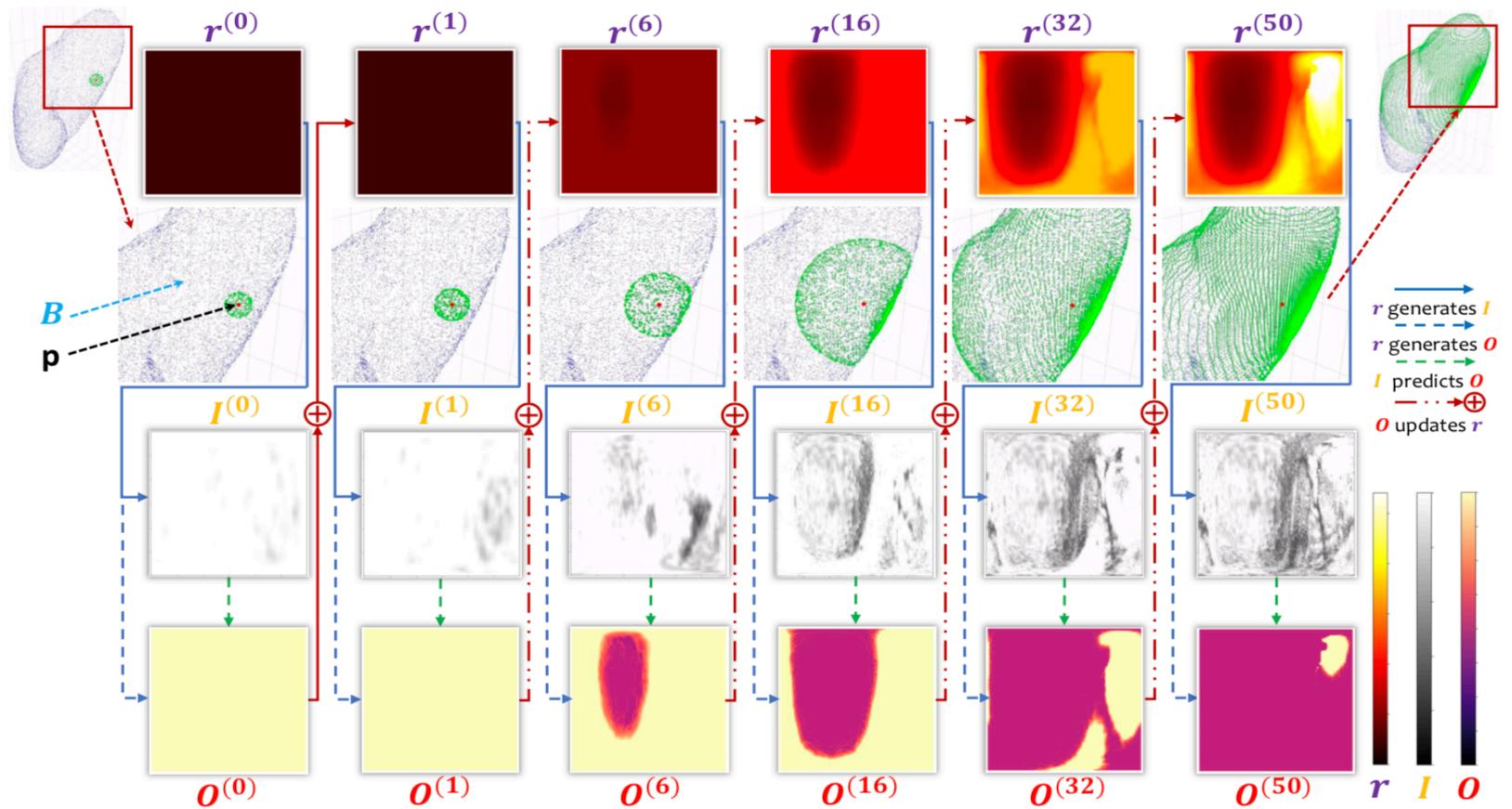


Figure 1. The overall flowchart of EBP (best viewed in color). We show the elastic shell after some specific numbers of iterations (green voxels in the second row) generated by a pivot p (the red center voxel in the second row) within an boundary \mathcal{B} of the organ (blue voxels in 2nd row). The data generation process starts from a perfect sphere initialized by $r^{(0)}$, and then we obtain the $(\mathbf{I}^{(t)}, \mathbf{O}^{(t)})$ pairs (the third and fourth row) by $r^{(t)}$ in the training stage. In the testing stage, $\mathbf{O}^{(t)}$ is predicted by our model M given $\mathbf{I}^{(t)}$. After that, one iteration is completed by the adjustment of $r^{(t)}$ to $r^{(t+1)}$ by the addition of $\mathbf{O}^{(t)}$. Finally, the elastic shell converges to \mathcal{B} .

EBP

- Set of *pivots* $P = \{p_1, p_2, \dots, p_N\}$
- Fixed set of *directions* $D = \{d_1, d_2, \dots, d_M\}$, d_m is a unit vector
$$\hat{x}_m^2 + \hat{y}_m^2 + \hat{z}_m^2 = 1.$$
- There is a *radius* $r_{\{n,m\}}$ indicating how far the boundary is along this direction, *i.e.*, $e_{\{n,m\}} = p_{\{n\}} + r_{\{n,m\}} \cdot d_m \in B$
- Volumetric segmentation reduces to the following problem: given a pivot p_n and a set of directions D , determine all $r_{\{n,m\}}$ so that $e_{\{n,m\}} = p_n + r_{\{n,m\}} \cdot d_m \in B$

Methods

- Given p_n , D and a group of $r_{\{n,m\}}$ values, determine whether these values correctly describe the boundary, *i.e.*, whether each $e_{n,m}$ falls on the boundary.
- Input: $I = U(p_n, \{d_m, r_{\{n,m\}}\}) = \{U(e_{\{n,m\}})\}$, where $U(e_{\{n,m\}})$ is the intensity value of U at position $e_{\{n,m\}}$
- Output: a map O of the same size, each value o_m indicating whether $e_{\{n,m\}}$ is located within, and how far it is from the boundary.

Methods

- Data Preparation: Distance to Boundary

$$|C(x, y, z)| = \min_{(x', y', z') \in \partial V} \text{Dist}[(x, y, z), (x', y', z')],$$

- After C is computed, they multiply C(x, y, z) by -1 for all background voxels, so that the sign of C(x, y, z) distinguishes inner voxels from outer voxels.
- Training: Data Generation and Optimization
 - spherical coordinate system, each direction has an azimuth angle $\alpha_m \in [0, 2\pi)$ and a polar angle $\phi_m \in [-\pi/2, \pi/2]$
 - They represent D as the Cartesian product of an azimuth angle set of M_a elements and a polar angle set of M_p elements

Methods

- Testing: Iteration and Inference
 - (1) the input image I_n only contains intensity values at the current shell,
 - (2) make use of the spatial consistency of distance prediction to improve accuracy.
 - (3) shrink the gap between training and testing data distributions
 - (4) construct a graph with all pivots being nodes and edges being connected between neighboring pivots.



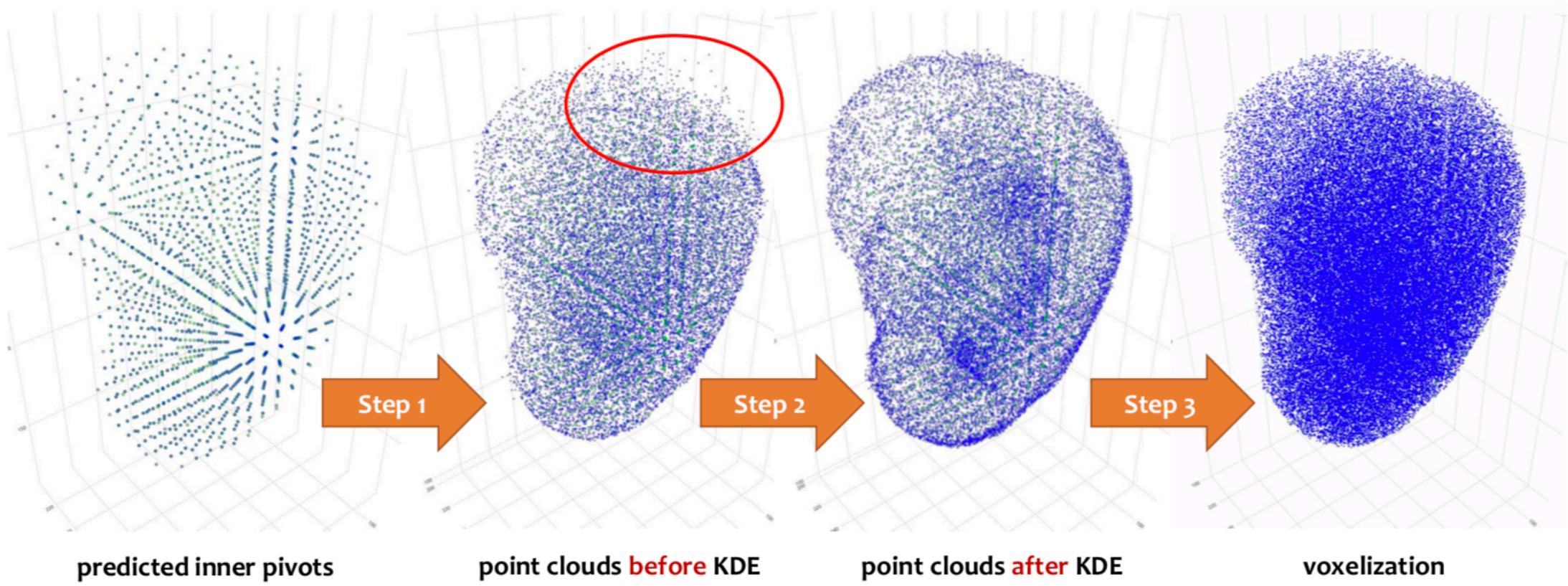


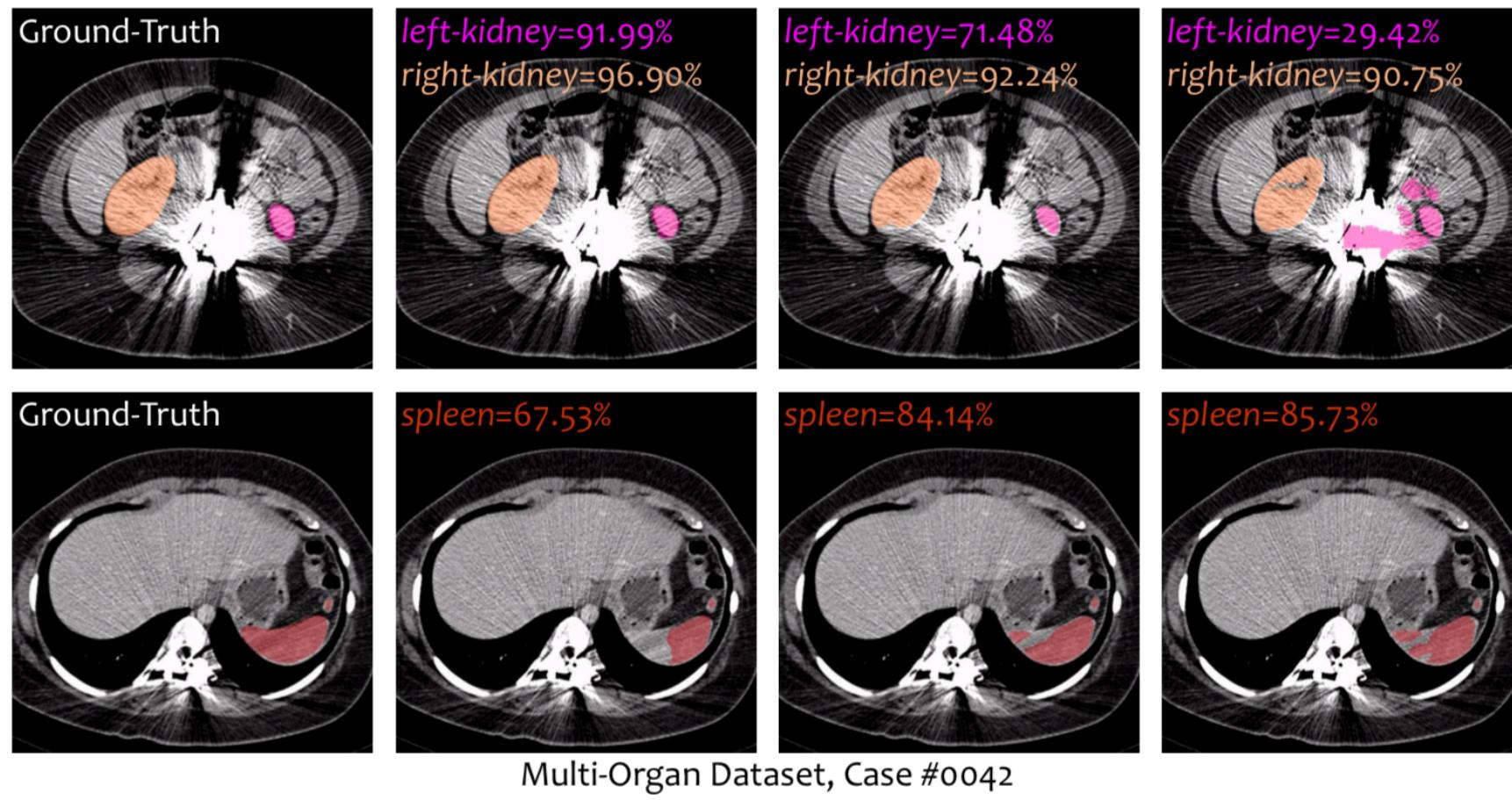
Figure 2. An example of 3D reconstruction (best viewed in color). We start with all pivots (green and blue points indicate ground-truth and predicted inner pivots, respectively) predicted to be located inside the target. In Step 1, all converged ending points generated by these pivots form the point clouds. In Step 2, a kernel density estimator (KDE) is applied to remove outliers (marked in a red oval in the second figure). In Step 3, we adopt a graphics algorithm for 3D reconstruction and finally we voxelize the point cloud.

Experiments

- Datasets:
 - 48 high-resolution CT scans, that were collected from some potential renal donors, and annotated by four expert radiologists
 - Four abdominal organs were labeled, including *left kidney*, *right kidney* and *spleen*
- Evaluation standards:
 - Dice coefficient



Visualizations

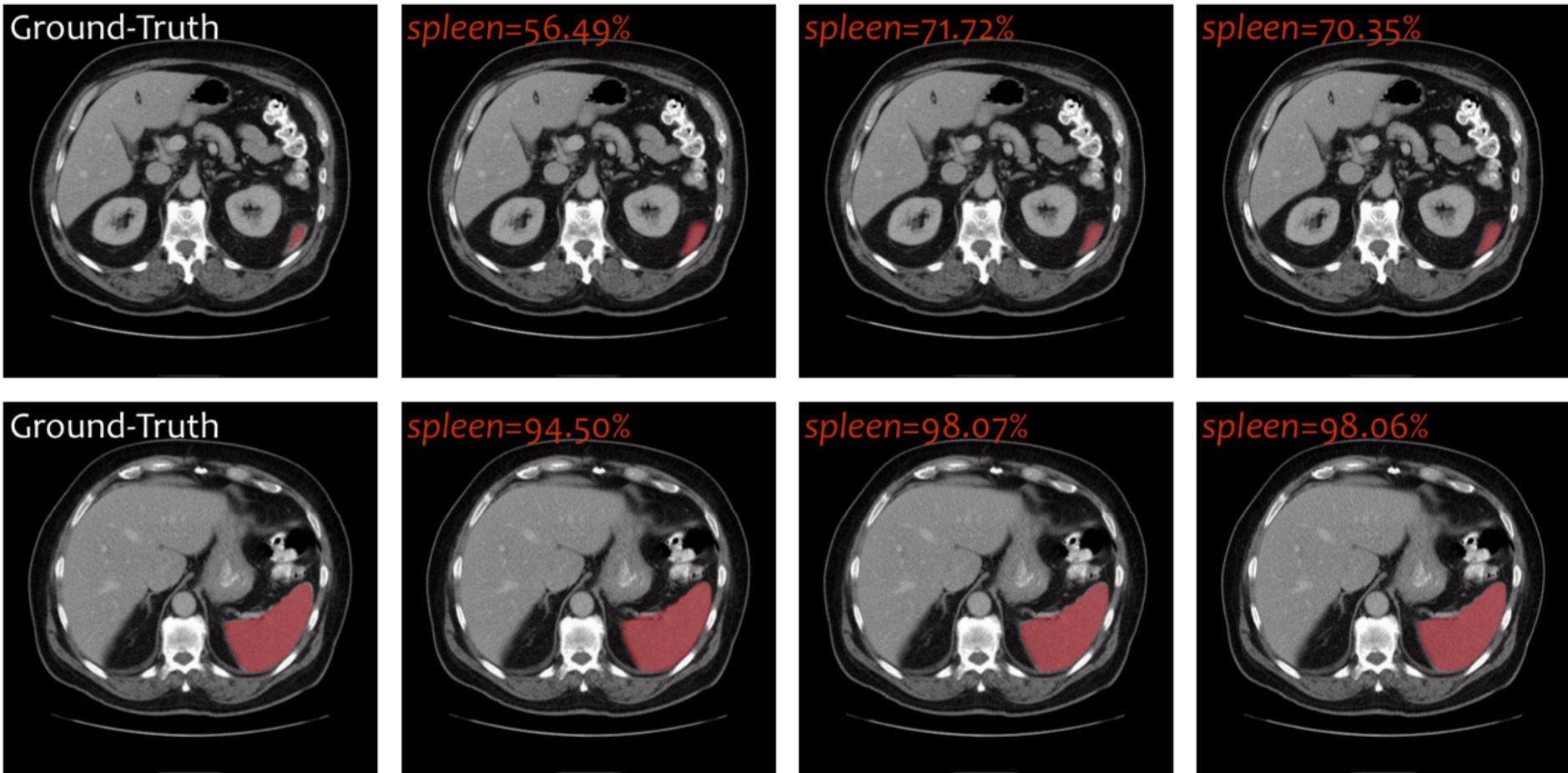


EBP Overall (3D) DSC: left-kidney=89.71%, right-kidney=93.70%, spleen=67.00%

RSTN Overall (3D) DSC: left-kidney=83.64%, right-kidney=87.35%, spleen=84.72%

VNet Overall (3D) DSC: left-kidney=71.40%, right-kidney=80.51%, spleen=83.18%

Visualizations



MSD spleen Dataset, Case #02

EBP Overall (3D) DSC=94.39%, **RSTN** Overall (3D) DSC=97.25%, **VNet** Overall (3D) DSC=86.88%

Results

Approach	<i>left kidney</i>			Approach	<i>right kidney</i>		
	Average	Max	Min		Average	Max	Min
RSTN [29]	94.50 ± 2.66	97.69	93.64	RSTN [29]	96.09 ± 2.21	98.18	87.35
VNet [20]	91.95 ± 4.63	95.23	71.40	VNet [20]	92.97 ± 3.67	97.48	80.51
EBP	93.45 ± 1.62	97.28	90.88	EBP	95.26 ± 1.59	97.45	90.19
Approach	<i>spleen</i>			Approach	MSD spleen		
	Average	Max	Min		Average	Max	Min
RSTN [29]	94.63 ± 4.21	97.38	78.75	RSTN [29]	89.70 ± 12.60	97.25	48.45
VNet [20]	92.68 ± 3.25	96.75	83.18	VNet [20]	92.94 ± 3.58	97.35	81.96
EBP	94.50 ± 2.64	96.76	89.67	EBP	92.01 ± 4.50	96.48	77.07

Table 2. Comparison of segmentation accuracy (DSC, %) on our multi-organ dataset and the *spleen* class in the MSD benchmark. Within each group, average (with standard deviation), max and min accuracies are reported.

Summary

- Overall:
 - A novel approach that trains 2D deep networks for 3D object segmentation
 - EBP is indeed equipped with the ability to locate the boundary
 - The core idea is to build up an elastic shell and adjust it until it converges to the actual boundary of the target
- Weakness:
- Future Work:

