End-to-End Learned Random Walker
for Seeded Image Segmentation

Supplementary Material

Lorenzo Cerrone, Alexander Zeilmann, Fred A. Hamprecht
Heidelberg Collaboratory for Image Processing
IWR, Heidelberg University, Germany
{lorenzo.cerrone, alexander.zeilmann, fred.hamprecht}@iwr.uni-heidelberg.de

1. Neural Network Architecture

In this section we present a schematic description of the CNNs we used in our approach (see Figures 1, 2 and 3).

All 3D convolutional layers use valid convolutions, except in the two output layers where we removed the padding in the z-axis in order to project the volume back to 2D. The filter size is written in each block and the number of filters below every block.

For downsampling (pointing down arrows), we used 3D maxpooling blocks with windows size of $1 \times 2 \times 2$. For the upsampling (pointing up arrows), we used 3D transpose convolutions with filters size of $1 \times 2 \times 2$.

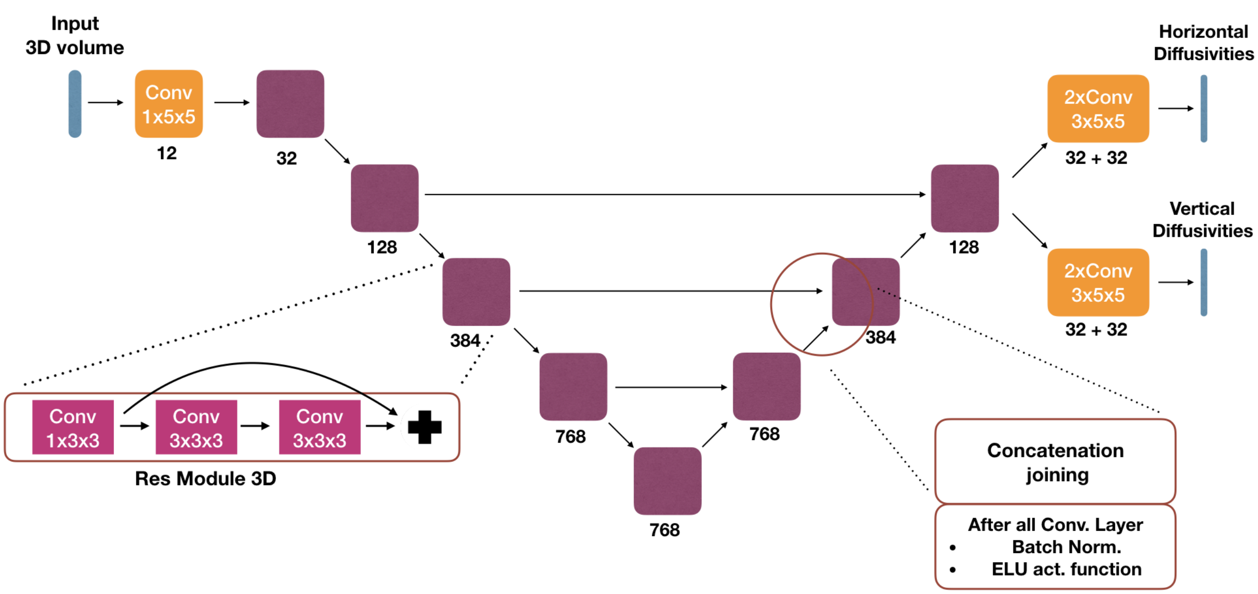

Figure 1. Illustration of the architecture used in our Learned Random Walker pipeline.

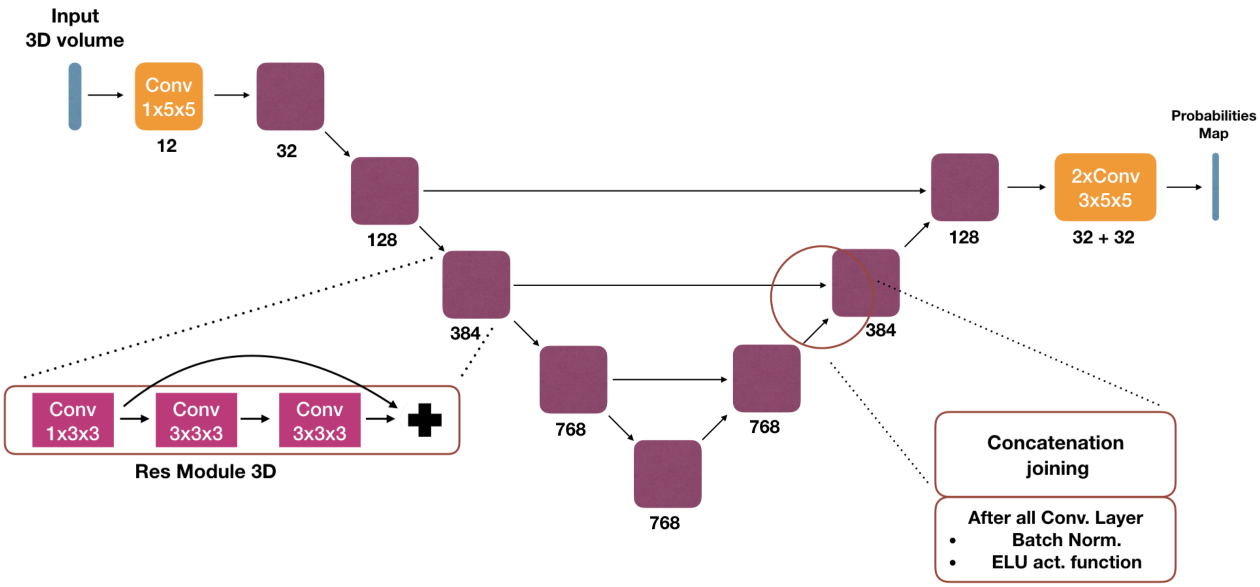

Figure 2. Illustration of the architecture used for our downsampled boundary probability map.

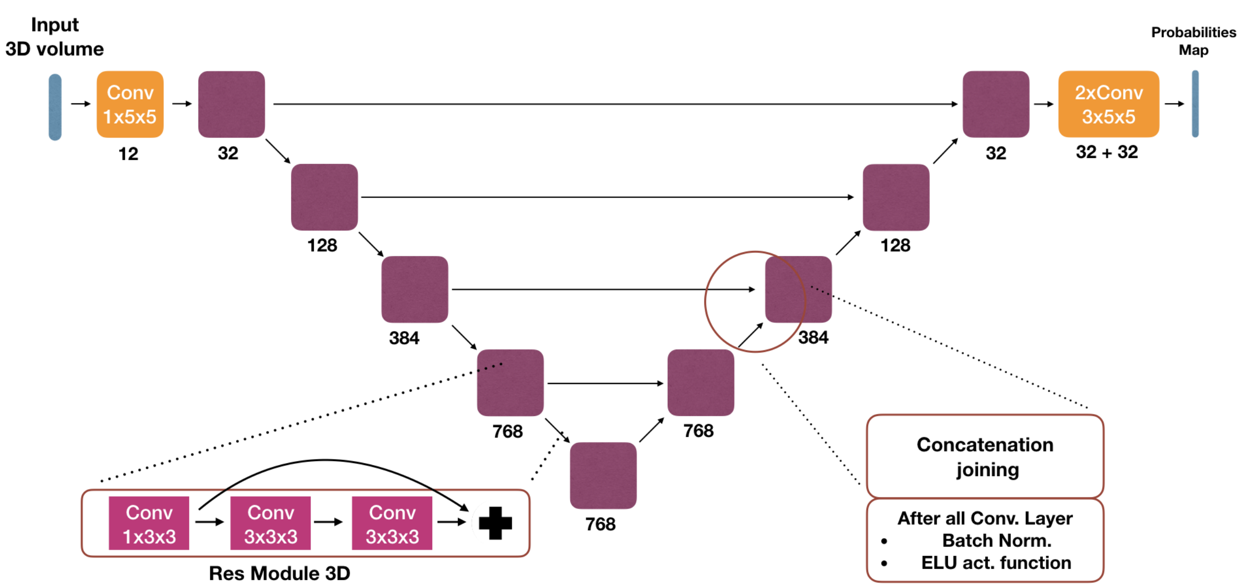

Figure 3. Illustration of the architecture used for our full size boundary probability map.

2. Sampling Strategy vs Approximate Back Propagation: Qualitative Results

In this section we present the results of our comparison with the approach of Vernaza and Chandraker.

Qualitatively, we can observe (see Figure 4) how all methods are capable of reconstructing the correct segmentation when dense seeding is provided.

On the other hand, the results are different when using sparse seeding (see Figure 5). In particular we can observe that the first order approximation fails in pixels far from any given seed.

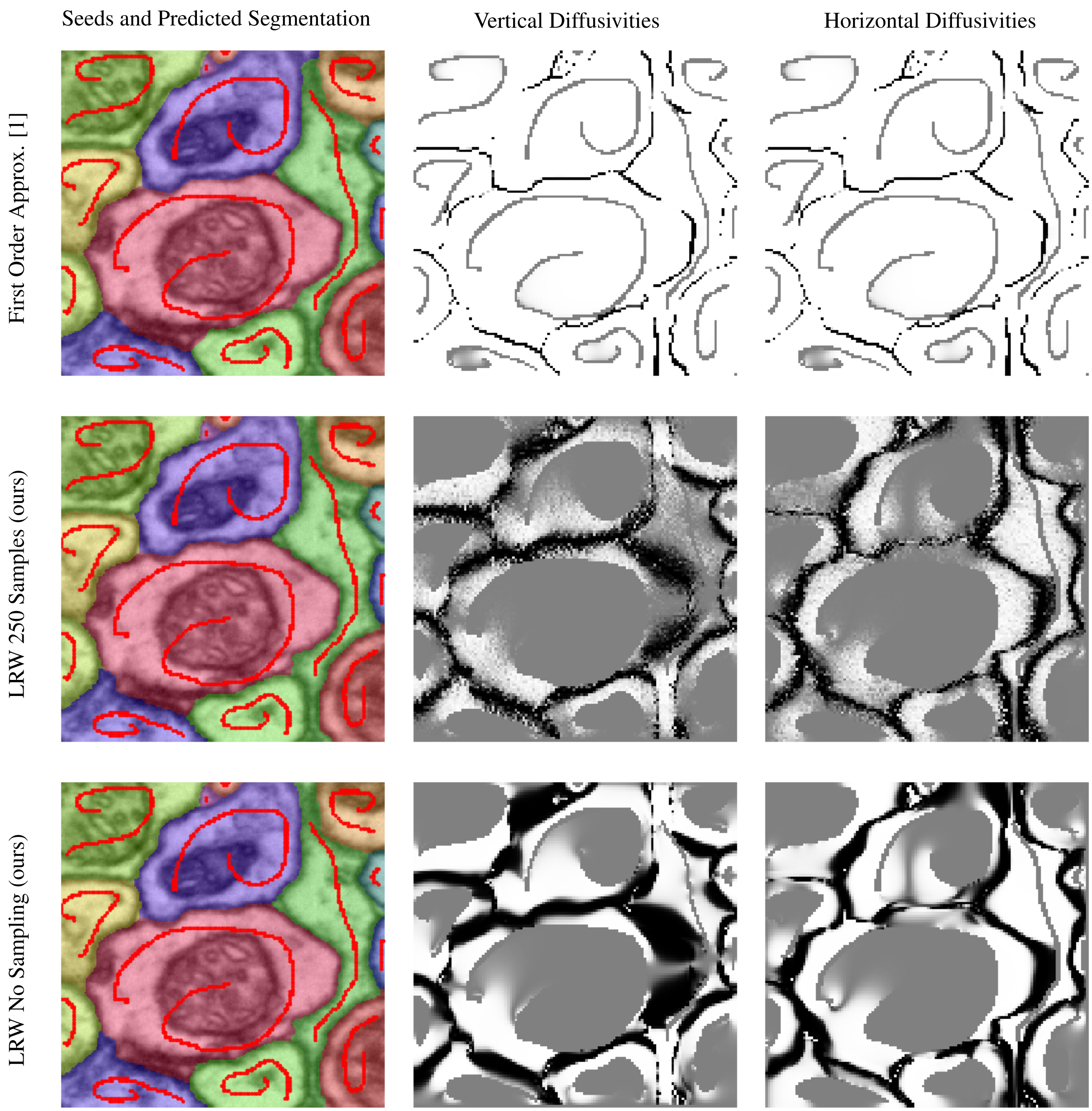

Figure 4. Summarized results of the comparison between Learned Random Walker and first order approximation [1]. The results are obtained with dense seeding.

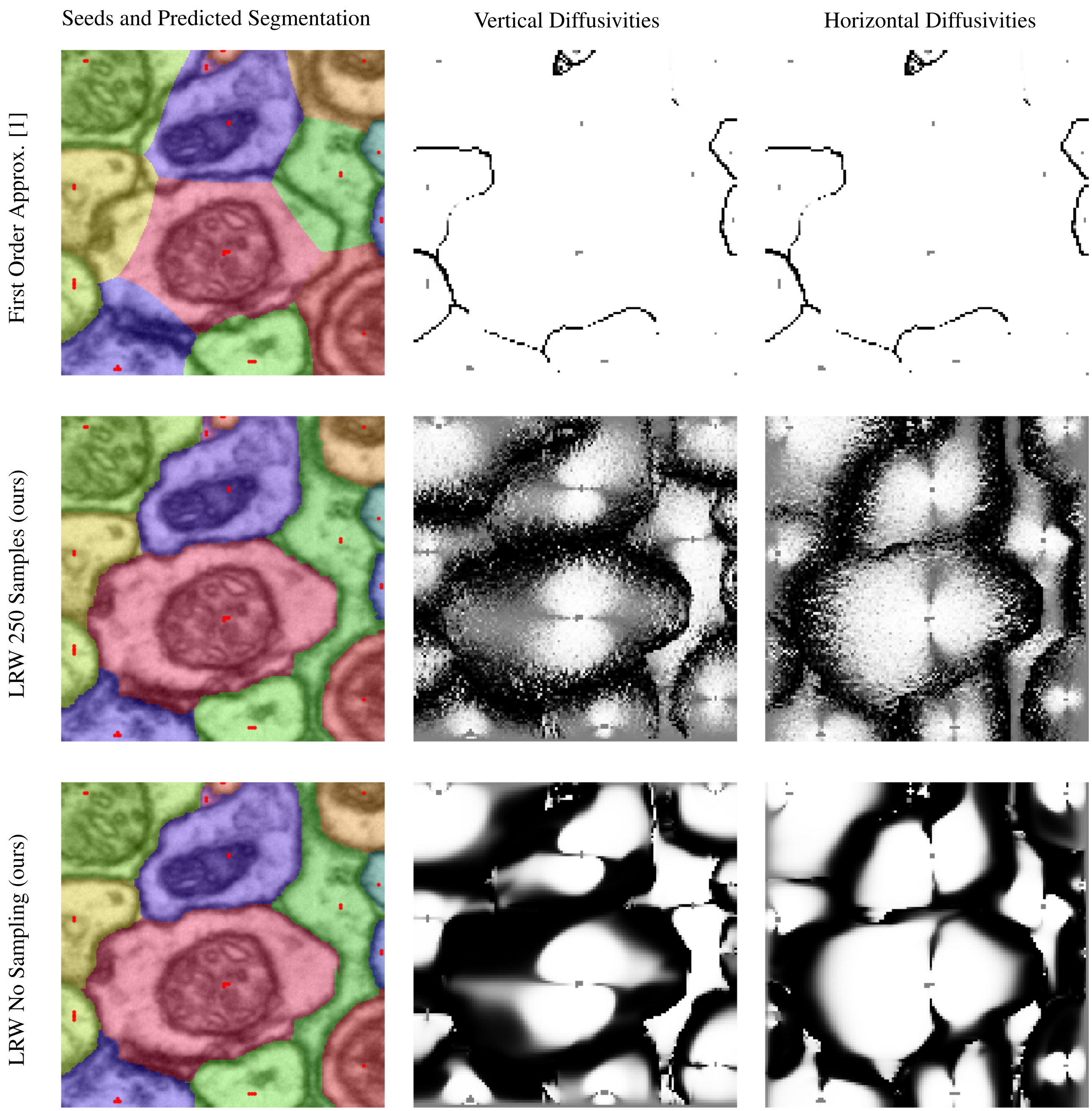

Figure 5. Summarized results of the comparison between Learned Random Walker and first order approximation [1]. The results are obtained with sparse seeding.

VOI	LRW with log barrier	LRW with side loss
CREMI A	0.076 ± 0.023	0.062 ± 0.021
CREMI B	0.220 ± 0.094	0.193 ± 0.089
CREMI C	0.272 ± 0.077	0.232 ± 0.081
Total	0.189 ± 0.109	0.162 ± 0.102

ARAND	LRW with log barrier	LRW with side loss
CREMI A	0.014 ± 0.077	0.011 ± 0.009
CREMI B	0.052 ± 0.053	0.045 ± 0.044
CREMI C	0.067 ± 0.036	0.061 ± 0.038
Total	0.044 ± 0.043	0.039 ± 0.040

Table 1. Quantitative comparison of the Learned Random Walker with log barrier and with side loss by looking at the means and standard deviations over the test set. Lower is better.

3. Purely Structured Training

In addition to the experiments we presented in the paper, we trained our Learned Random Walker pipeline without any side loss. In its place, we used a log barrier on the edge weights as an unsupervised regularization.
With this, the new loss function reads:

$$J\left(Z^*, Z, \Theta\right) = \mathrm{CE}(Z^*, Z, \Theta) - \frac{\alpha}{2|V|}\left\|\log(w)\right\|_1 + \frac{\beta}{2}\left\|\Theta\right\|_2^2. \tag{1}$$

The terms are weighted by $\alpha = 10^{-5}$ and $\beta = 10^{-5}$.
The results obtained with this setup are presented in Table 1. Despite the errors being larger without the side loss, the scores with the side loss are still competitive.

4. Further Results for the CREMI Challenge

In this section we present the strongest and weakest results of our Learned Random Walker for the CREMI challenge.

4.1. Strongest Results

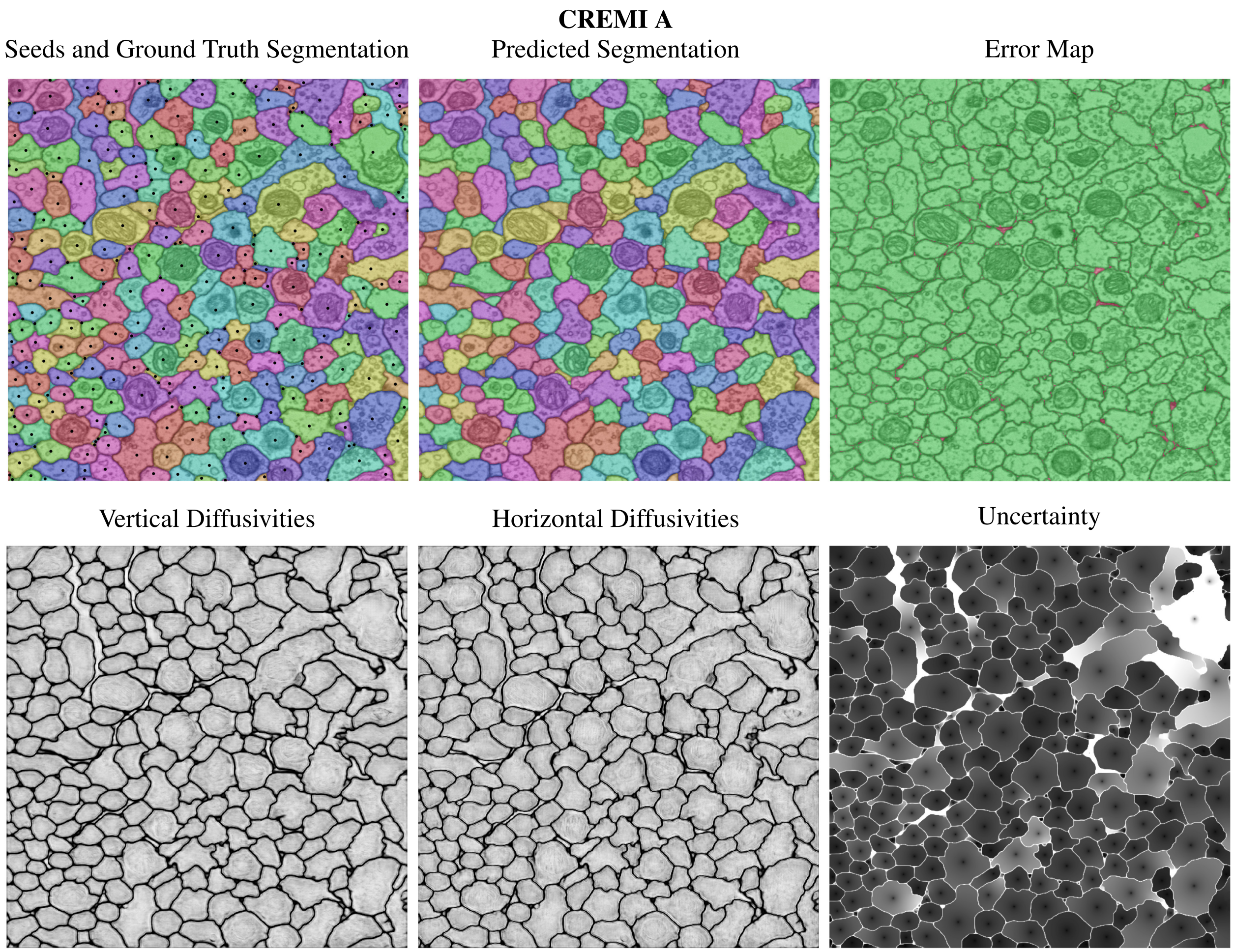

Figure 6. **CREMI A**, slice 17, ARAND $= 0.003$, VOI $= 0.027$.

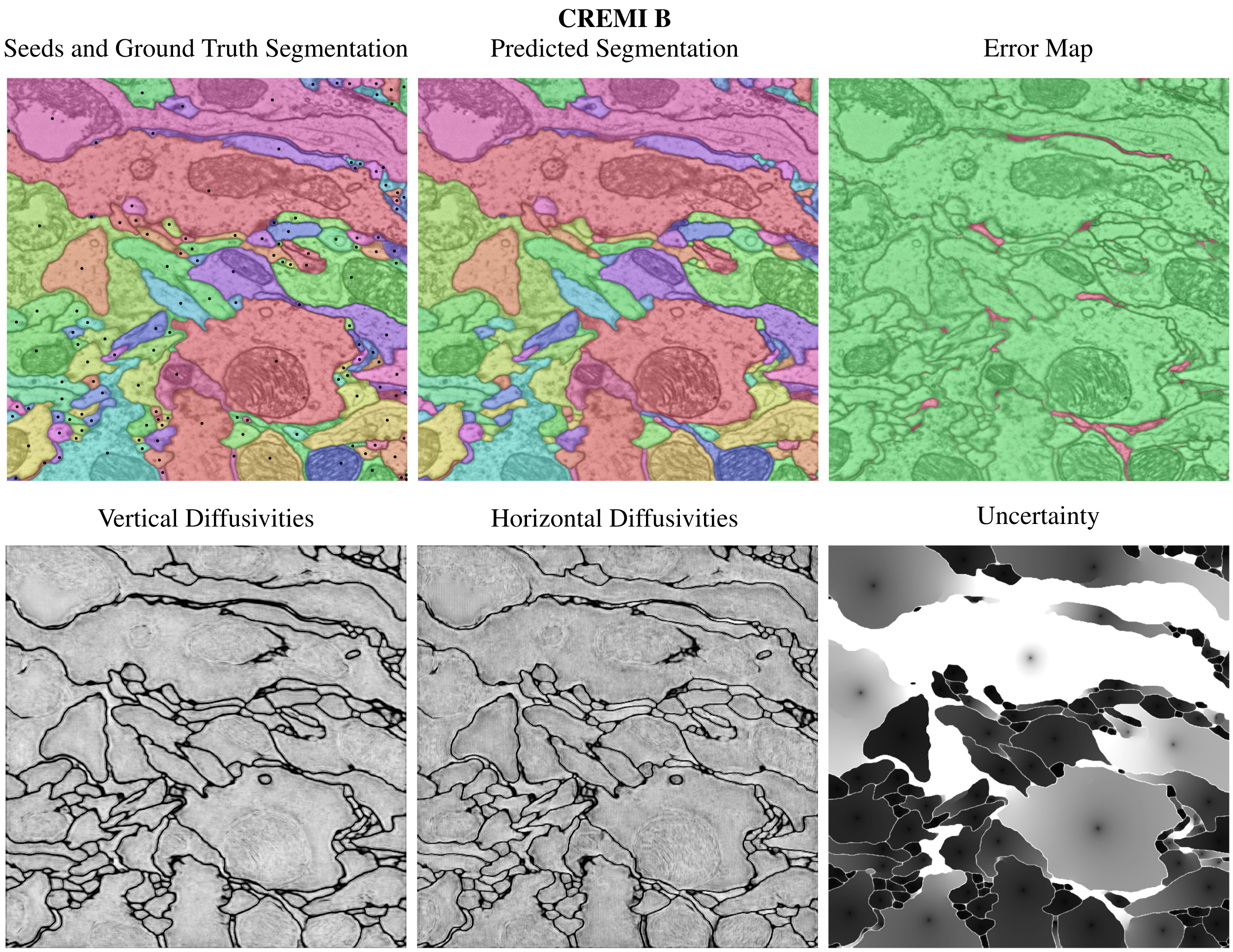

Figure 7. **CREMI B**, slice 29, ARAND $= 0.006$, VOI $= 0.095$.

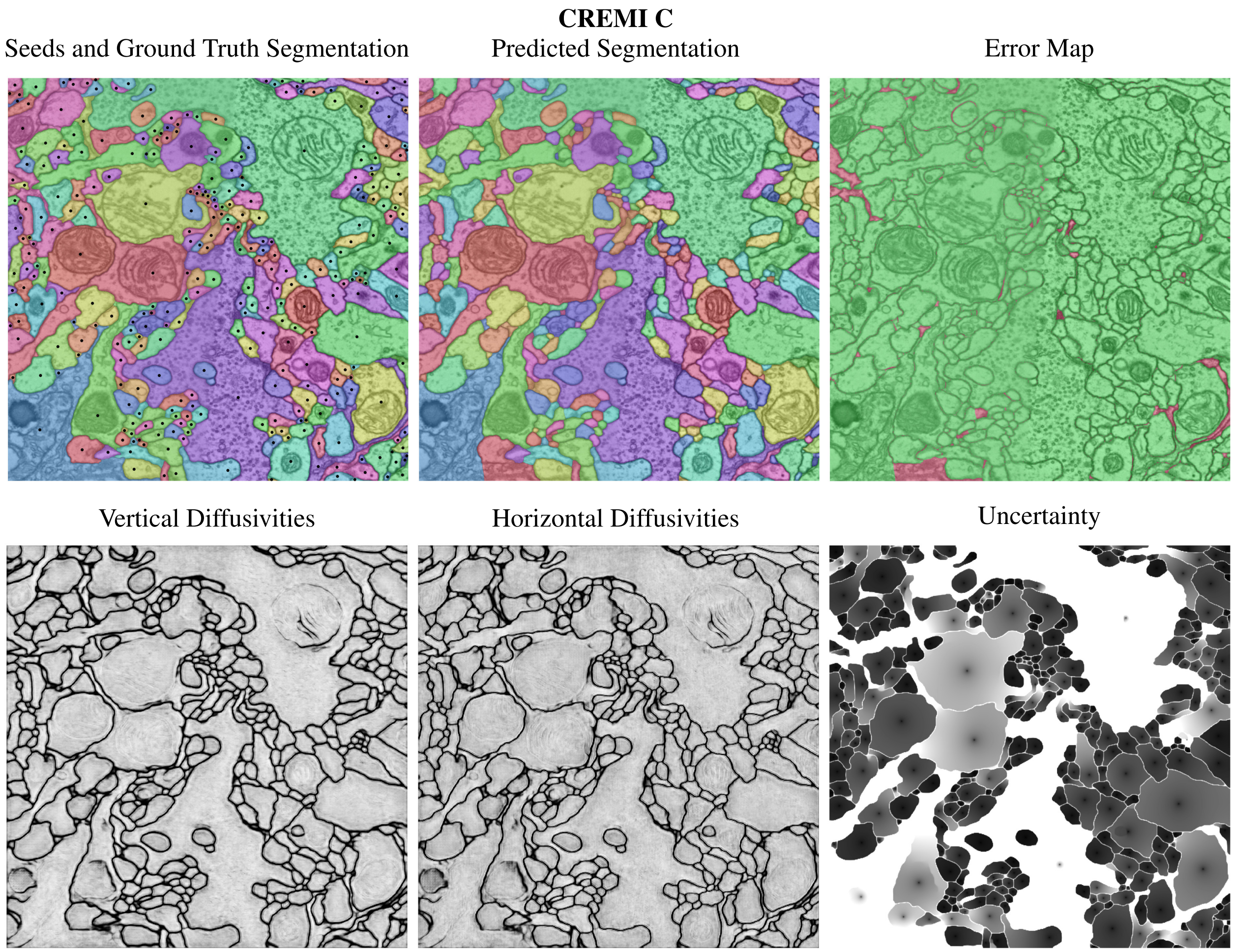

Figure 8. **CREMI C**, slice 49, ARAND $= 0.015$, VOI $= 0.148$.

4.2. Weakest Results

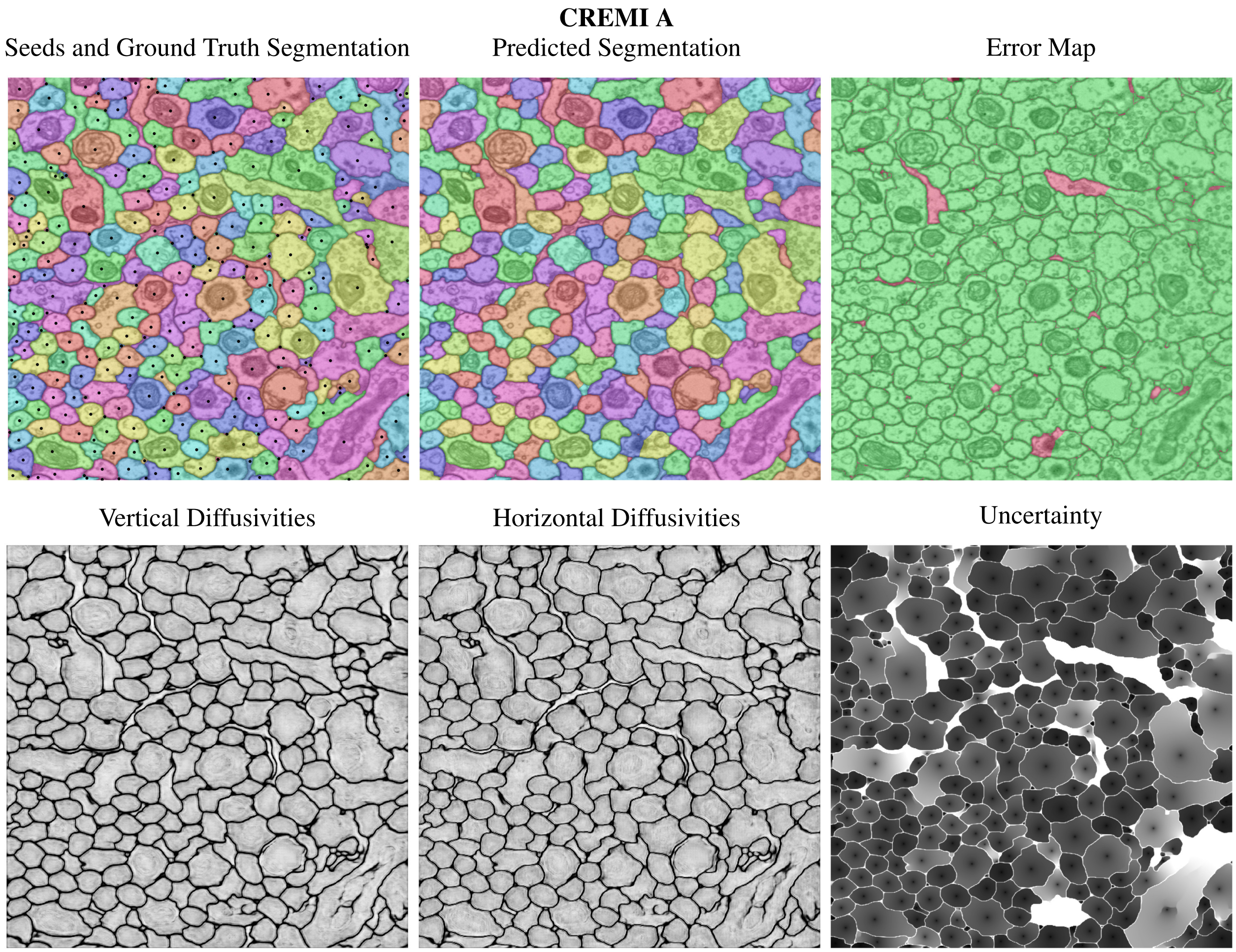

Figure 9. **CREMI A**, slice 5, ARAND $= 0.041$, VOI $= 0.116$.

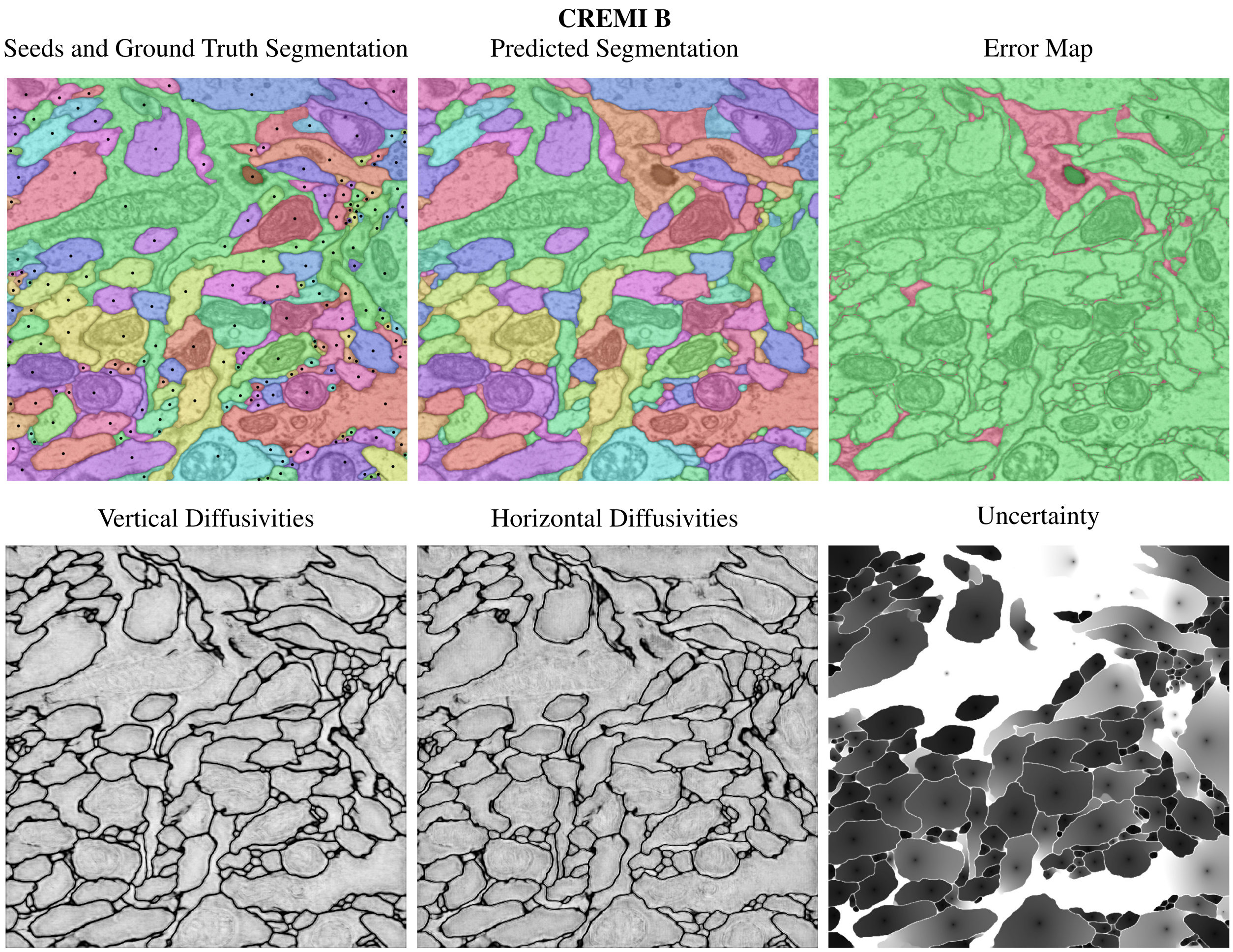

Figure 10. **CREMI B**, slice 4, ARAND = 0.168, VOI = 0.319.

CREMI C

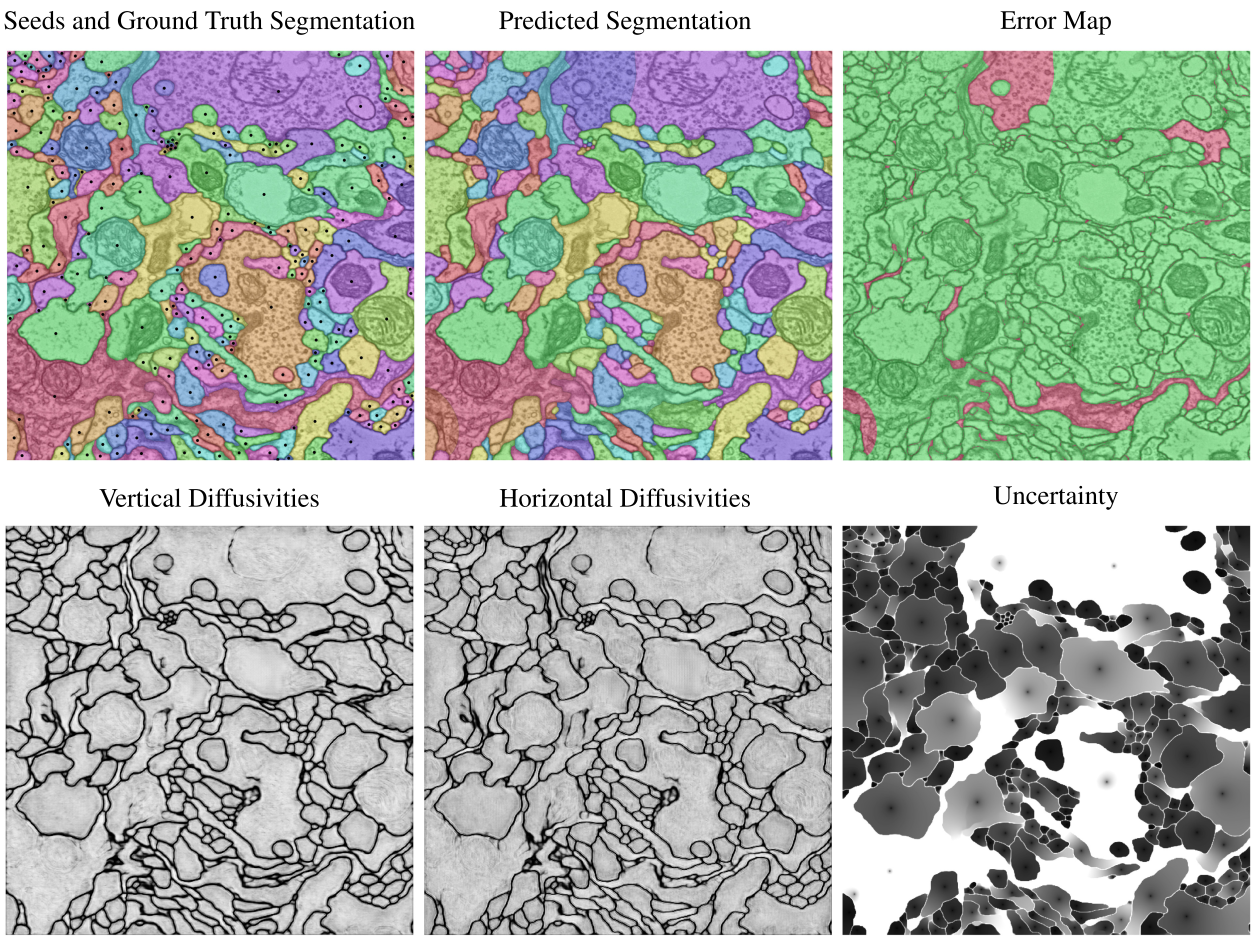

Figure 11. **CREMI C**, slice 32, ARAND $= 0.181$, VOI $= 0.401$.

References

[1] Paul Vernaza and Manmohan Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 00, pages 2953–2961, July 2017.