Efficient Context-Aware Network for Abdominal Multi-organ Segmentation

Fan Zhang[1], Yu Wang[2], Hua Yang[3]
[1, 2, 3]Department of Radiation Algorithm, Fosun Aitrox
Information Technology Company. Shanghai, China.
zhangfan2@fosun.com[1], wangyu@fosun.com[2], yanghua@fosun.com[3]

Abstract

The contextual information, presented in abdominal CT scan, is relative consistent. In order to make full use of the overall 3D context, we develop a whole-volume-based coarse-to-fine framework for efficient and effective abdominal multi-organ segmentation. We propose a new ***efficientSegNet*** *network, which is composed of encoder, decoder and context block. For the decoder module, anisotropic convolution with a k*k*1 intra-slice convolution and a 1*1*k inter-slice convolution, is designed to reduce the computation burden. For the context block, we propose strip pooling module to capture anisotropic and long-range contextual information, which exists in abdominal scene. Quantitative evaluation on the FLARE2021 validation cases, this method achieves the average dice similarity coefficient (DSC) of 0.895 and average normalized surface distance (NSD) of 0.775. This method won the 1st place on the 2021-MICCAI-FLARE challenge. Codes and models are available at https://github.com/Shanghai-Aitrox-Technology/EfficientSegmentation*

1. Introduction

In this paper, we focus on multi-organ segmentation from abdominal CT scans. As shown in Figure 1, the main difficulties stem from four aspects: 1) The variations in field-of-views, shape and size of different organs. 2) The abnormalities, like lesion-affected organ, may lead to segmentation failure. 3) The diversity of data source in term of multi-center, multi-phase and multi-vendor cases. 4) The limited GPU memory size and high computation cost.

A common solution [1] is to develop a sliding-window method, which can balance the GPU memory usage. Usually, this method need to sample sub-volumes overlap with each other to improve the segmentation accuracy, while leading to more computation cost. Meanwhile, sub-volumes sampled from entire CT volume inevitably lose some 3D context, which is important for distinguishing multi-organ with respect to background.

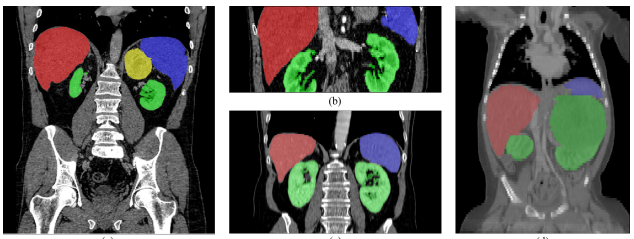

Figure 1. An illustration of abdominal CT images vary in field-of-views (a-c), and lesion-affected organ (d) on FLARE2021 dataset.

We develop a whole-volume-based coarse-to-fine framework [2] to effectively and efficiently tackle these challenges. The coarse model aims to obtain the rough location of target organ from the whole CT volume. Then, the fine model refines the segmentation based on the coarse result. This coarse-to-fine pipeline can cover anatomical variations for different cases. To capture the spatial relationships between multi-organ, we exploit strip pooling [3] for collecting anisotropic and long-range context. This strip pool offers two advantages. Firstly, compared to self-attention or non-local module, strip pool consumes less memory and matrix computation. Secondly, it deploys long but narrow pooling kernels along one spatial dimension to simultaneously aggregate both global and local context.

The main contributions of this work are summarized as follows:

1) We propose a whole-volume-based coarse-to-fine framework to make full use of the overall 3D context, this pipeline covers the anatomical variations occurred on different cases.

2) We design anisotropic convolution block with low computation cost. We propose strip pooling module to capture anisotropic and long-range contextual information.

3) The effectiveness and efficiency of the proposed whole-volume-based coarse-to-fine framework are demonstrated on FLARE2021 challenge dataset, where we achieve the state-of-the-art with low time cost and less memory usage.

2. Method

As mentioned in Figure 2, this whole-volume-based coarse-to-fine framework is composed of coarse and fine segmentation. A detail description of the method is as follows.

2.1. Preprocessing

The baseline method includes the following preprocessing steps:

- Reorientation image to target direction.
- Resampling image to fixed size. Coarse input: [160, 160, 160]. Fine input: [192, 192, 192].
- Intensity normalization: First, the image is clipped to the range [-325, 325]. Then a z-score normalization is applied based on the mean and standard deviation of the intensity values.

2.2. Proposed Method

The proposed **efficientSegNet** consists of three major parts: the feature encoder module, the context extractor module, and the feature decoder module, as shown in Figure 3.

As depicted in Figure 4 and Figure 5, the encoder module is composed of two residual convolution blocks, and the decoder module with one residual convolution block. As to decoder module, we separate a standard 3D convolution with kernel size 3×3×3 into a 3×3×1 intra-slice convolution and a 1×1×3 inter-slice convolution. The residual convolution block is implemented as follows: conv-instnorm-ReLU-conv-instnorm-ReLU (where the addition of the residual takes place before the last ReLU activation).

We adopt 3D-based mixed pyramid pooling (Figure 6) to extract contextual feature, which is composed of the standard spatial pooling and the anisotropic strip pooling. The standard spatial pooling employs two average pooling with the stride of 2×2×2 and 4×4×4. The anisotropic strip pooling with three different-direction receptive fields: 1×N×N, N×1×N and N×N×1, where N is the size of feature map in last encoder module.

The initial number of feature maps is 8 for coarse model, while 16 for fine model. We aggregate low and high level feature with addition rather than concatenation, because the former consumes less GPU memory. In addition, the number of model parameters is 9 MB, and the number of flops is 333 GB for 192×192×192 input size.

2.3. Post-processing

A connected component analysis of segmentation mask is applied on coarse and fine model output.

3. Dataset and Evaluation Metrics

3.1. Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [4] (Liver [5], Spleen, Pancreas), NIH Pancreas [6, 7, 8], KiTS [9, 10], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [11]. The detail information is presented in Table 1.

- Details of training / validation / testing splits:
  The total number of cases is 511. An approximate 70%/10%/20% train/validation/testing split is employed resulting in 361 training cases, 50 validation cases, and 100 testing cases. The detail information is presented in Table 1.

- Furthermore, the training dataset (361 cases) is randomly divided into training (80%) and validation (20%) set, where validation set is used to model selection. A 5-fold cross validation set is generated based on the above mentioned partition.

3.2. Evaluation Metrics

- Dice Similarity Coefficient (DSC)
- Normalized Surface Distance (NSD)
- Running time
- Maximum used GPU memory (when the inference is stable)

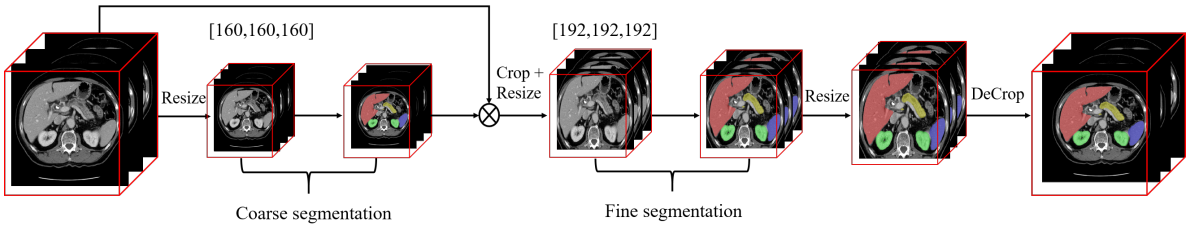

Figure 2. A schematic diagram of whole-volume-based coarse-to-fine segmentation framework.

Table 1. Data splits of FLARE2021.

Data Split	Center	Phase	# Num.
Training (361 cases)	The National Institutes of Health Clinical Center	portal venous phase	80
	Memorial Sloan Kettering Cancer Center	portal venous phase	281
Validation (50 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
Testing (100 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
	Nanjing University	various phases	50

Table 2. Environments and requirements.

Ubuntu version	16.04.12
CPU	Intel(R) Xeon(R) Gold 5118 CPU @ 2.30GHz (×4)
RAM	502 GB
GPU	Nvidia GeForce 2080Ti (×8)
CUDA version	10.1
Programming language	Python3.6
Deep learning framework	Pytorch (torch 1.5.0, torchvision 0.2.1)
Code is publicly available at	EfficientSegmentation

Table 3. Training protocols.

Data augmentation methods	Crop and brightness.
Initialization of the network	Kaiming normal initialization
Patch sampling strategy	Augment the sample ratio of pathological image (3 times)
Batch size	16
Patch size	Coarse: 160×160×160 Fine: 192×192×192
Total epochs	200
Optimizer	Adam with betas (0.9, 0.99), L2 penalty: 0.00001
Loss	Dice loss and focal loss (alpha = 0.5, gamma = 2)
Dropout rate	0.2
Initial learning rate	0.01
Learning rate decay schedule	Step decay
Stopping criteria, and optimal model selection criteria	Stopping criterion is reaching the maximum number of epoch (200).
Training mode	Mixed precision
Training time for coarse model	3 hours
Training time for fine model	6 hours

4. Implementation Details

4.1. Environments and requirements

The environments and requirements of the proposed method is shown in Table 2.

4.2. Training protocols

The training protocols of the proposed method is shown in Table 3.

4.3. Testing protocols

The same pre-process and post-process methods are applied as training steps. In order to reduce the time cost of pre-process and post-process, resample and intensity normalization are computed in GPU. We implement the connected component analysis in C++ library, namely

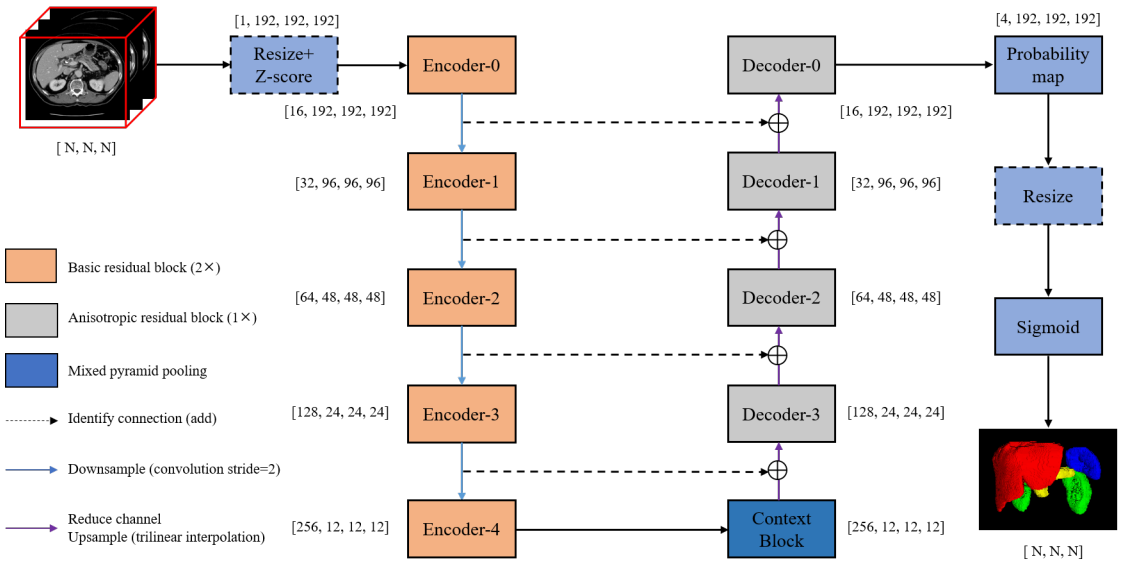

Figure 3. Illustration of the proposed efficientSegNet.

Table 4. Quantitative results of 5-fold cross validation in terms of DSC and NSD.

Training	Liver		Kidney		Spleen		Pancreas	
	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)
Fold-1	96.8±5.6	88.2±11.4	95.0±8.4	89.6±12.7	95.9±13.9	93.5±15.1	79.3±23.4	68.7±23.7
Fold-2	96.3±6.6	87.4±11.3	94.0±11.0	88.3±14.3	95.7±11.5	93.2±13.5	79.5±22.0	68.9±22.2
Fold-3	96.6±5.1	88.1±10.3	94.8±9.9	89.1±13.0	95.6±13.9	93.1±15.3	78.3±23.0	68.0±23.1
Fold-4	96.3±6.9	88.3±11.1	95.8±5.4	89.5±12.4	95.5±14.5	93.3±16.1	80.9±21.7	69.2±21.4
Fold-5	96.5±6.2	87.1±11.7	94.5±10.5	88.7±14.4	95.8±11.6	93.5±13.3	79.5±20.9	66.7±21.7
Average	96.5±6.1	87.8±11.2	94.8±9.3	89.0±13.4	95.7±13.1	93.3±14.7	79.5±22.2	68.3±22.5

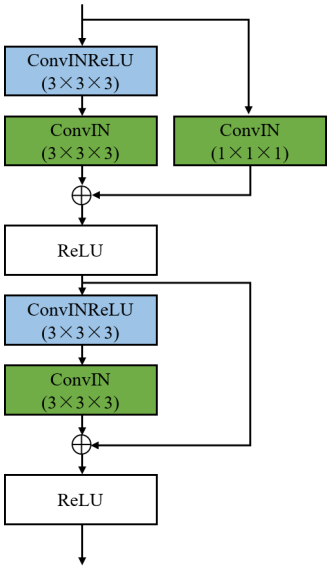

Figure 4. Illustration of the encoder block.

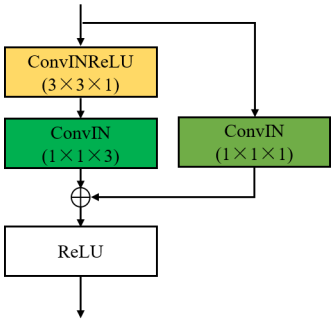

Figure 5. Illustration of the decoder block.

cc3d [12]. We implement the inference model in FP16 mode. Dynamic empty cache is used to reduce GPU memory.

5. Results

5.1. Quantitative results for 5-fold cross validation.

The provided results analysis is based on the 5-fold cross validation results on training set. Table 4 illustrates the results of 5-fold cross validation. While high DSC and NSD scores are obtained for liver, kidney and spleen, DSC

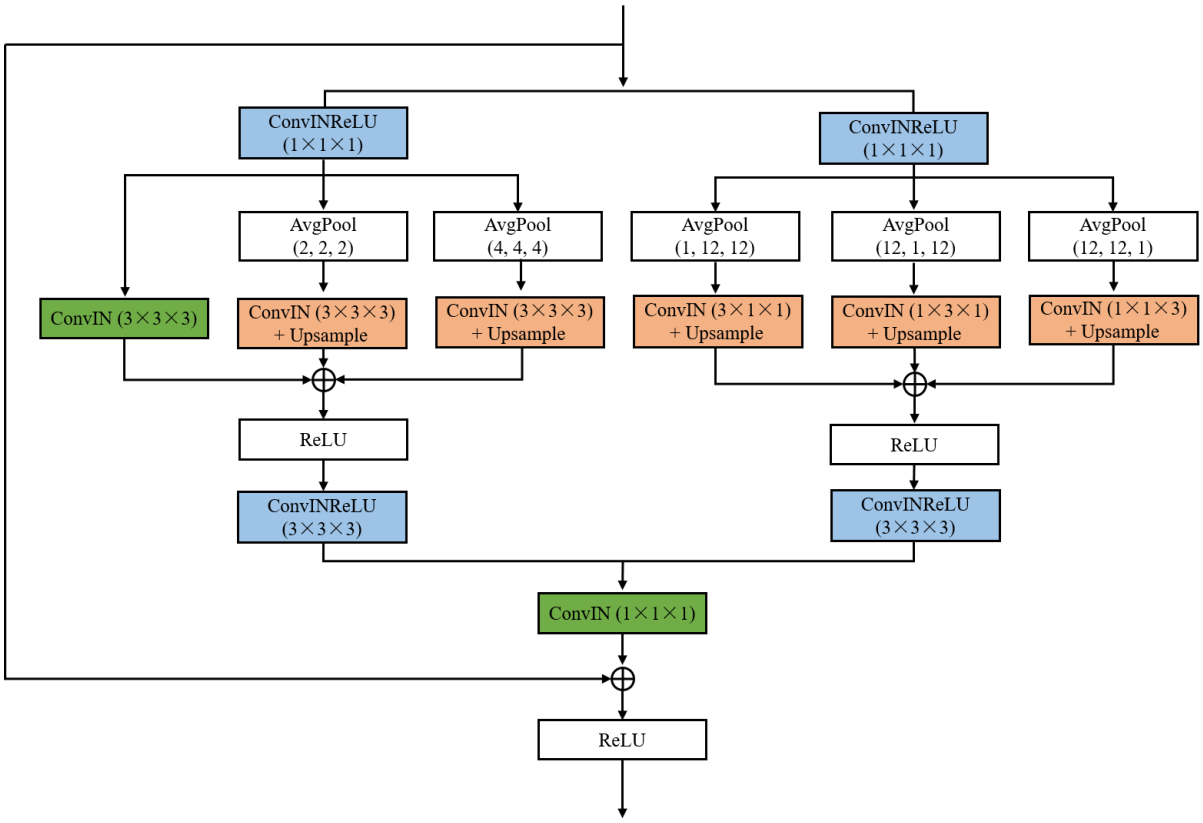

Figure 6. Illustration of the context block.

and NSD scores for pancreas indicating unsatisfactory performance.

5.2. Quantitative results on validation set.

The average running time is 9.8 s per case in inference phase, and maximum used GPU memory is 1017 MB. Table 5 illustrates the results on validation set. Compared to the 5-fold cross validation on training set, the results degrade of DSC (approximately 1 point for liver, kidney and spleen, 4 point for Pancreas organ) on validation set is relative low, which indicates highly generation ability. While the results degrade of NSD (approximately 7 point) on validation set is fairly obvious, demonstrating that the boundary regions contain more segmentation errors, which need further improvements.

5.3. Qualitative results

Figure 7 presents some easy and hard examples on validation set, and quantitative result is illustrated in Table 6. It can be found that the proposed method can segment healthy (case #15) or slightly lesion-affected (case #47) organs well, while disappointing performance on seriously lesion-affected (case #23 and #25) organs.

Table 5. Quantitative results of validation set in terms of DSC and NSD.

Organ	DSC (%)	NSD (%)
Liver	95.4±6.60	80.4±13.77
Kidney	93.6±6.50	82.8±12.28
Spleen	94.2±13.88	87.1±16.66
Pancreas	75.3±17.44	60.5±16.66
Average	89.65	77.75

6. Discussion and Conclusion

The proposed method can work well on cases where healthy or slightly lesion-affected organs. The proposed method achieves the highly generation ability for liver, kidney and spleen segmentation in terms of DSC scores. Disappointing performance is obtained for pancreas segmentation as a result of the inter-patient anatomical variability of volume and shape. The existence of seriously lesion-affected organ is a critical factor for the poor segmentation performance. Besides, obtaining an accurate boundary segmentation need further investigate.

Table 6. The DSC and NSD scores of easy and hard examples.

Example	Liver		Kidney		Spleen		Pancreas	
	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)
Case #15	98.0	91.6	97.8	94.5	98.2	97.7	88.3	79.3
Case #47	98.7	93.7	95.8	87.9	97.8	94.2	92.3	80.3
Case #23	**56.3**	**40.4**	91.6	66.3	96.4	82.6	74.8	50.3
Case #25	84.3	71.1	96.4	87.7	97.4	95.3	**13.5**	**14.1**

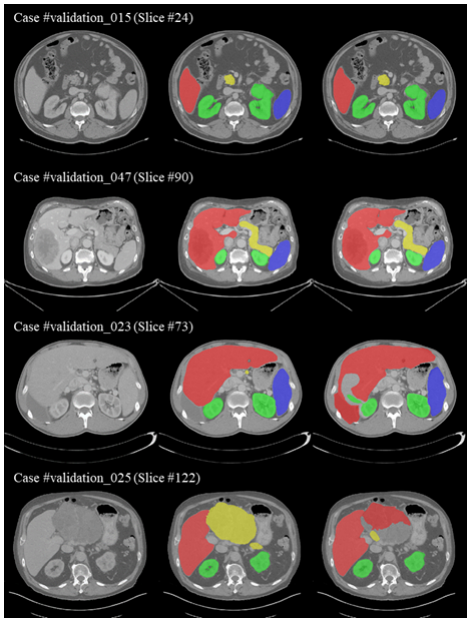

Figure 7. Qualitative results on easy (case #15 and #47) and hard (case #23 and #25) examples. First column is the image, second column is the ground truth, and third column is the predicted results by our propose method.

Acknowledgment

We sincerely appreciate the organizers with the donation of FLARE2021 dataset. We declare that pre-trained models and additional datasets are not used in this paper.

References

[1] F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, "nnu-net: a self-configuring method for deep learning-based biomedical image segmentation," *Nature Methods*, vol. 18, no. 2, pp. 203–211, 2021. 1

[2] Z. e. a. Zhu, "A 3d coarse-to-fine framework for volumetric medical image segmentation." *2018 International Conference on 3D Vision (3DV)*, 2018. 1

[3] Q. e. a. Hou, "Strip pooling: Rethinking spatial pooling for scene parsing.." *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[4] A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. Van Ginneken, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," *arXiv preprint arXiv:1902.09063*, 2019. 2

[5] P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser *et al.*, "The liver tumor segmentation benchmark (lits)," *arXiv preprint arXiv:1901.04056*, 2019. 2

[6] H. Roth, A. Farag, E. Turkbey, L. Lu, J. Liu, and R. Summers, "Data from pancreas-ct. the cancer imaging archive (2016)." 2

[7] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2015, pp. 556–564. 2

[8] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle *et al.*, "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, no. 6, pp. 1045–1057, 2013. 2

[9] N. Heller, F. Isensee, K. H. Maier-Hein, X. Hou, C. Xie, F. Li, Y. Nan, G. Mu, Z. Lin, M. Han *et al.*, "The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge," *Medical Image Analysis*, vol. 67, p. 101821, 2021. 2

[10] N. Heller, S. McSweeney, M. T. Peterson, S. Peterson, J. Rickman, B. Stai, R. Tejpaul, M. Oestreich, P. Blake, J. Rosenberg *et al.*, "An international challenge to use artificial intelligence to define the state-of-the-art in kidney and kidney tumor segmentation in ct imaging." *American Society of Clinical Oncology*, vol. 38, no. 6, pp. 626–626, 2020. 2

[11] J. Ma, Y. Zhang, S. Gu, C. Zhu, C. Ge, Y. Zhang, X. An, C. Wang, Q. Wang, X. Liu, S. Cao, Q. Zhang, S. Liu, Y. Wang, Y. Li, J. He, and X. Yang, "Abdomenct-1k: Is abdominal organ segmentation a solved problem?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[12] W. Silversmith., "cc3d: Connected components on multilabel 3d images." January 2021. [Online]. Available: https://github.com/seung-lab/connected-components-3d/ 4