

Portable framework to deploy deep learning segmentation models for medical images

Aditi Iyer*, Eve Locastro*, Aditya P. Apte**†, Harini Veeraraghavan, Joseph O. Deasy

Department of Medical Physics, Memorial Sloan Kettering Cancer Center, New York, NY 10065 USA.

*These authors contributed equally to this work.

† Corresponding author: apte@mskcc.org

Purpose: This work presents a framework for deployment of deep learning image segmentation models for medical images across different operating systems and programming languages.

Methods: Computational Environment for Radiological Research (CERR) platform was extended for deploying deep learning-based segmentation models to leverage CERR's existing functionality for radiological data import, transformation, management, and visualization. The framework is compatible with MATLAB as well as GNU Octave and Python for license-free use. Pre and post processing configurations including parameters for pre-processing images, population of channels, and post-processing segmentations was standardized using JSON format. CPU and GPU implementations of pre-trained deep learning segmentation models were packaged using Singularity containers for use in Linux and Conda environment archives for Windows, macOS and Linux operating systems. The framework accepts images in various formats including DICOM and CERR's planC and outputs segmentation in various formats including DICOM RTSTRUCT and planC objects. The ability to access the results readily in planC format enables visualization as well as radiomics and dosimetric analysis. The framework can be readily deployed in clinical software such as MIM via their extensions.

Results: The open-source, GPL copyrighted framework developed in this work has been successfully used to deploy Deep Learning based segmentation models for five in-house developed and published models. These models span various treatment sites (H&N, Lung and Prostate) and modalities (CT, MR). Documentation for their usage and demo workflow is provided at <https://github.com/cerr/CERR/wiki/Auto-Segmentation-models>. The framework has also been used in clinical workflow for segmenting images for treatment planning and for segmenting publicly available large datasets for outcomes studies.

Conclusions: This work presented a comprehensive, open-source framework for deploying deep learning-based medical image segmentation models. The framework was used to translate the developed models to clinic as well as reproducible and consistent image segmentation across institutions, facilitating multi-institutional outcomes modeling studies.

Keywords: image segmentation, deep-learning, radiomics, radiotherapy outcomes, normal tissue complication, tumor control, model inference.

Introduction:

This work fills the need for a portable software framework useful to deploy segmentation models for medical images. There are several general-purpose frameworks [PyTorch¹, Tensorflow², Caffe³, Keras⁴] for training deep learning-based models. There are also frameworks to deploy and distribute inference models such as DeepInfer⁵, ModelHub⁶ and NiftyNET⁷. DeepInfer, which can also be invoked from 3DSlicer⁸, currently provides segmentation models for Prostate gland, biopsy needle trajectory and tip and brain white matter hyperintensities. NiftyNET⁷ is an open-source platform to train and deploy TensorFlow-based convolutional neural networks (CNN) models. It provides the implementation of networks such as HighRes3DNet, 3D U-net, V-net and DeepMedic. It can be used to train new models as well as share pre-trained models. While these generic frameworks offer a wide variety of network architectures, they were not designed to support specific processing requirements of medical images and porting of models between users. The framework developed in this work provides end-to-end workflow for using deep learning-based image segmentation models including conversion between data formats, visualization and simplified access to image metadata. This enables users to readily apply the trained models on datasets collected at different institutions as well as incorporate the validated models in clinical workflow.

Methods:

The framework was developed as a module within the Computational Environment for Radiological Research⁹ (CERR) software platform. This allows users to leverage existing CERR functionalities such as flexible planC data structure, visualization, feature extraction and ability

to use in various programming languages including Matlab, Octave and Python via the Oct2Py bridge. Deep learning Model and framework dependencies must be consistent to ensure reproducibility of segmentation models. Container technology such as Docker and Singularity allow users to securely bundle libraries such that they are compatible with a variety of scientific computing architectures. Singularity container technology presents various advantages in terms of usage on bare-metal HPC clusters including no requirements of root privileges. While Singularity container technology completely isolates the environment from the host, it has limitations for use on Windows and Mac operating systems. Conda is a popular cross platform package and environment manager that installs and manages conda packages from the Anaconda repository as well as from the Anaconda Cloud. Conda environments can be archived and installed on other systems and locations using tools such as Conda-Pack. This is useful for deploying code in a consistent environment—potentially on systems where python and/or conda isn't already installed. CPU and GPU implementations of pre-trained deep learning segmentation models were packaged using Singularity containers (Linux) and Conda

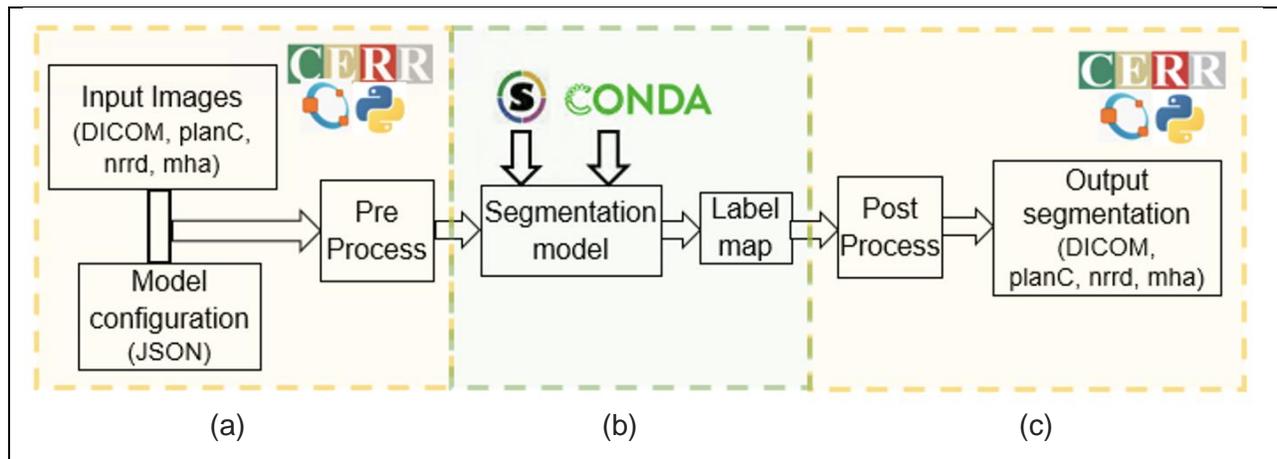


Figure 1. Overview of deep learning segmentation framework. (a) CERR is used in a license-free Octave or Python programming language to pre-process input images per model configuration. (b) Segmentation models are distributed as packaged Conda environments for specific operating systems (Windows, Mac, Linux) or Singularity containers for Linux. (c) Resulting segmentation labels are post-processed in CERR per model configuration.

environment archives (Windows, macOS).

Several pre-processing operations specific to the medical imaging domain are supported, including automated cropping to patient outline, 2D/3D resampling, 2D/2.5D/3D and multimodal channels, and orientation transformations (transverse, sagittal, coronal). The framework supports models derived with multiple image modalities. E.g. MR and CT. Various image registration options include deformable and rigid registrations as wrapper calls to popular packages such as Plastimatch¹⁰ and ANTS¹¹. Additionally, images can be processed using Image Biomarker Standardization¹² Initiative defined filters. Various post-processing operations are supported, including filtering to retain selected number of connected components by size or within a user-defined region of interest. Dynamic label-to-structure name maps are supported where the number output structures are variable. Table 1 lists the available processing operations. Specification of processing operations is simplified by defining them via JSON configuration file per model.

Table 1. Supported pre- and post-processing methods

OPERATION	SUPPORTED METHODS
Resampling	Sinc, linear, cubic, spline, makima, nearest.
Cropping	Crop to: input margin, bounding-box (2D/3D), selected no. slices, around patient body (2D/3D) or apply custom method.
Resizing	Zero padding (2D/3D), bilinear, bicubic, sinc.
Transforming orientation	Transverse, sagittal, coronal
Modality	Single/multiple (CT,MR,PT)
Image filters	Haralick, Laws, Wavelet, Sobel, Gabor, LoG, CoLIAGe.
Population of channels	2D, 2.5D, 3D
Post-processing of label maps	Filters to return largest connected component(s) by size or within region of interest, custom methods.

Results:

The segmentation framework is distributed as an open-source, GNU-copyrighted software with the CERR platform whereas the developers of models retain the copyright to their models. Documentation for available models and their usage is provided at <https://github.com/cerr/CERR/wiki/Auto-Segmentation-models>. A Jupyter notebook runnable on Google Colab platform demonstrates the installation of various dependencies their usage - https://github.com/cerr/CT_SwallowingAndChewing_DeepLabV3/blob/master/demo_DLseg_swallowing_and_chewing_structures.ipynb. The framework has been used to share deep learning segmentation models with several institutions within and outside the USA. Additionally, it has been used in clinical segmentation and facilitating outcomes studies¹³ using large external/open datasets.

Table 2 Available pre-trained models

SITE	STRUCTURES	FRAMEWORK	MODALITY
Lung	Heart, Pericardium, Atria, Ventricles	DeepLab, Pytorch (Haq et al ¹⁴)	CT
Prostate	Bladder, Prostate and Seminal Vesicles (CTV), Penile Bulb, Rectum, Urethra and Rectal Spacer	DeepLabV3+, Tensorflow (Elguindi et al ¹⁵)	MR
H&N	Parotids (left, right), Submandibulars (left, right), Mandible and Brain Stem	Self-Attention, Pytorch (Jiang et al)	CT
Lung	Nodules	Incremental MRRN, Keras, Tensorflow (Jiang et al ¹⁶)	CT
H&N	Masseters (left, right), medial pterygoids (left, right), larynx, and constrictor muscle	DeepLabV3+, Pytorch (Iyer et al)	CT
Brain	Brain metastases	Vnet, Tensorflow (Hsu et al)	CT, MR

Acknowledgements

This research was partially funded by NIH grant 1R01CA198121 and NIH/NCI Cancer Center Support grant P30 CA008748.

References

- 1 Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library", (2019).
- 2 Martin and Barham Abadi, Paul and Chen, Jianmin and Chen, Zhifeng and Davis, Andy and Dean, Jeffrey and Devin, Matthieu and Ghemawat, Sanjay and Irving, Geoffrey and Isard, Michael and others, "Tensorflow: A system for large-scale machine learning", in *Symposium on Operating Systems Design and Implementation* (2016), pp. 265-283.
- 3 Yangqing and Shelhamer Jia, Evan and Donahue, Jeff and Karayev, Sergey and Long, Jonathan and Girshick, Ross and Guadarrama, Sergio and Darrell, Trevor, "Caffe: Convolutional Architecture for Fast Feature Embedding," arXiv preprint arXiv:1408.5093 (2014).
- 4 Francois and others Chollet, (GitHub, 2015).
- 5 A. Mehrtash, M. Pesteie, J. Hetherington, P. A. Behringer, T. Kapur, W. M. Wells, 3rd, R. Rohling, A. Fedorov, and P. Abolmaesumi, "DeepInfer: Open-Source Deep Learning Deployment Toolkit for Image-Guided Therapy," *Proc SPIE Int Soc Opt Eng* **10135** (2017).
- 6 Michael Schwier Ahmed Hosny, Christoph Berger, Evin P Örneke, Mehmet Turan, et al, "ModelHub.AI: Dissemination Platform for Deep Learning Models," arXiv preprint arXiv:1911.13218 (2019).
- 7 E. Gibson, W. Li, C. Sudre, L. Fidon, D. I. Shkir, G. Wang, Z. Eaton-Rosen, R. Gray, T. Doel, Y. Hu, T. Whyntie, P. Nachev, M. Modat, D. C. Barratt, S. Ourselin, M. J. Cardoso, and T. Vercauteren, "NiftyNet: a deep-learning platform for medical imaging," *Comput Methods Programs Biomed* **158**, 113-122 (2018).
- 8 Pieper SD Kikinis R, Vosburgh K, "3D Slicer: A Platform for Subject-Specific Image Analysis, Visualization, and Clinical Support," In: Jolesz F. (eds) *Intraoperative Imaging and Image-Guided Therapy* (2014).
- 9 A. P. Apte, A. Iyer, M. Crispin-Ortuzar, R. Pandya, L. V. van Dijk, E. Spezi, M. Thor, H. Um, H. Veeraraghavan, J. H. Oh, A. Shukla-Dave, and J. O. Deasy, "Technical Note: Extension of CERR for computational radiomics: A comprehensive MATLAB platform for reproducible radiomics research," *Med Phys* (2018); J. O. Deasy, A. I. Blanco, and V. H. Clark, "CERR: a computational environment for radiotherapy research," *Med Phys* **30** (5), 979-985 (2003).

- 10 P. Zaffino, P. Raudaschl, K. Fritscher, G. C. Sharp, and M. F. Spadea, "Technical Note: plastimatch mabs, an open source tool for automatic image segmentation," *Med Phys* **43** (9), 5155 (2016).
- 11 B. B. Avants, N. J. Tustison, J. Wu, P. A. Cook, and J. C. Gee, "An open source multivariate framework for n-tissue segmentation with evaluation on public data," *Neuroinformatics* **9** (4), 381-400 (2011).
- 12 A. Zwanenburg, M. A. Abdalah, A. Apte, S. Ashrafinia, J. Beukinga, M. Bogowicz, C. V. Dinh, M. Götz, M. Hatt, R. T. H. Leijenaar, J. Lenkowicz, O. Morin, A. U. K. Rao, J. Socarras Fernandez, M. Vallières, L. V. Van Dijk, J. Van Griethuysen, F. H. P. Van Velden, P. Whybra, E. G. C. Troost, C. Richter, and S. Löck, "PO-0981: Results from the Image Biomarker Standardisation Initiative," *Radiotherapy and Oncology* **127**, S543-S544 (2018).
- 13 M. Thor, A. Apte, R. Haq, A. Iyer, E. LoCastro, and J. O. Deasy, "Using Auto-Segmentation to Reduce Contouring and Dose Inconsistency in Clinical Trials: The Simulated Impact on RTOG 0617," *Int J Radiat Oncol Biol Phys* **109** (5), 1619-1626 (2021).
- 14 R. Haq, A. Hotca, A. Apte, A. Rimner, J. O. Deasy, and M. Thor, "Cardio-pulmonary substructure segmentation of radiotherapy computed tomography images using convolutional neural networks for clinical outcomes analysis," *Phys Imaging Radiat Oncol* **14**, 61-66 (2020).
- 15 S. Elguindi, M. J. Zelefsky, J. Jiang, H. Veeraraghavan, J. O. Deasy, M. A. Hunt, and N. Tyagi, "Deep learning-based auto-segmentation of targets and organs-at-risk for magnetic resonance imaging only planning of prostate radiotherapy," *Phys Imaging Radiat Oncol* **12**, 80-86 (2019).
- 16 J. Jiang, Y. C. Hu, C. J. Liu, D. Halpenny, M. D. Hellmann, J. O. Deasy, G. Mageras, and H. Veeraraghavan, "Multiple Resolution Residually Connected Feature Streams for Automatic Lung Tumor Segmentation From CT Images," *IEEE Trans Med Imaging* **38** (1), 134-144 (2019).