Sensitivity Analysis for Interpretation of Machine Learning Based Segmentation Models in Cardiac MRI

Markus J Ankenbrand (markus.ankenbrand@uni-wuerzburg.de)
University Hospital Würzburg https://orcid.org/0000-0002-6620-807X
Liliia Shainberg
University Hospital Würzburg
Michael Hock
University Hospital Würzburg
David Lohr
University Hospital Würzburg
Laura M Schreiber
University Hospital Würzburg

Software

Keywords: Deep learning, neural networks, cardiac magnetic resonance, sensitivity analysis, transformations, augmentation, segmentation

Posted Date: October 29th, 2020

DOI: https://doi.org/10.21203/rs.3.rs-97535/v1

Version of Record: A version of this preprint was published on February 15th, 2021. See the published version at https://doi.org/10.1186/s12880-021-00551-1.

Abstract

Background Image segmentation is a common task in medical imaging e.g. for volumetry analysis in cardiac MRI. Artificial neural networks are used to automate this task with performance similar to manual operators. However, this performance is only achieved in the narrow tasks networks are trained on. Performance drops dramatically when data characteristics differ from the training set properties. Moreover, neural networks are commonly considered black boxes, because it is hard to understand how they make decisions and why they fail. Therefore, it is also hard to predict whether they will generalize and work well with new data.

Results We show that sensitivity analysis is a suitable approach to answer practical questions regarding use and functionality of segmentation models. We also provide an open source Python library (misas), that facilitates the use of this method with arbitrary data and models. By enabling a better understanding of neural networks through sensitivity analysis it also assists in decision making. We demonstrate this in two case studies on cardiac magnetic resonance imaging.

Conclusions Sensitivity analysis is a useful tool for deep learning developers as well as users such as clinicians. It extends their toolbox with a new tool that makes segmentation models more interpretable. Although demonstrated only on cardiac magnetic resonance images this approach and software are much more broadly applicable.

Background

Image segmentation is of great interest in medical imaging, e.g. in imaging of tumors (1, 2), retina (3), lung (4), and the heart (5). In the latter, segmentation is applied to partition acquired images into functionally meaningful regions. Quantitative static and dynamic measures of diagnostic relevance are derived from that. These measures include myocardial mass, ventricular volumes, wall thickness, wall motion and ejection fraction. State-of-the-art performance for automatic segmentation is achieved with artificial neural networks (6–8). Many researchers demonstrated impressive performance on their test task and target data. However, neural networks also have limitations, mainly regarding generalization to new data and interpretability (9).

The limited generalization is particularly problematic as both training data and real world data are rarely from the exact same distribution. Methods to deal with so-called data set shift are subject of ongoing research (10). Furthermore, there might be the effect of hidden stratification (11), there is usually some kind of bias in sampling the training data (12) and networks might learn shortcuts (13) using unintended features to boost performance on the training set. This is commonly addressed by using diverse data sources and extensive data augmentation or sophisticated models (14). A general framework to evaluate, quantify and boost generalization is missing.

Explainability and interpretability of neural networks are additional active fields of research (9, 15). In model interpretability the goal is to understand how and why a model makes certain predictions. While

local interpretability describes a certain prediction by the model based on a defined input, global interpretability delineates the understanding of general features determining the models' predictions. Specifically for neural networks a variety of methods have been recently developed to determine so called attribution (16). Here attribution means evaluating the contribution of input features (17), layers (18) or single neurons (19) to the prediction.

Sensitivity analysis was first proposed by Widrow et al. in the context of misclassification caused by weight perturbations because of noisy input and machine imprecision (20). Ever since the term sensitivity analysis has been overloaded with different meanings related to each other. There is an entire book about sensitivity analysis in neural networks dealing with sensitivity to parameter noise (21). Here we define sensitivity analysis as exploration of the effect of input transformations on model predictions. The most closely related approach to the one presented here uses algorithm sensitivity analysis for tissue image segmentation (22). This work shares the general idea, however, differs in a variety of factors such as automatic parameter search and its focus on computational performance (22).

In this work, we describe a straightforward method to interpret arbitrary segmentation models. This sensitivity analysis provides intuitive local interpretations by transforming an input image in a defined manner and inspecting the impact of that transformation on the model performance.

It can be used to answer common questions in machine learning projects: Can a network, trained and published by someone else, be applied to my own data? Is it necessary or beneficial to prepare the data in a certain way? We demonstrate how these questions can be addressed by sensitivity analysis in the first case study. Other common questions are: How robust is a model that was trained on a limited dataset regarding characteristics of the data (e.g. orientation, brightness)? How problematic are potential perturbations such as image artifacts? An approach to solve this issue is described in the second case study.

Beside describing the method and highlighting its utility in two case studies, in addition we present an open source python library called misas (model interpretation through sensitivity analysis for segmentation) that makes it easy to apply sensitivity analysis to new data and segmentation models.

Implementation

The software library described in this article is written in Python 3. The development was achieved by literate programming (23) in Jupyter notebooks using the nbdev framework, which provides all library code, documentation, and tests in one place. The source code is hosted on GitHub (https://github.com/chfc-cmi/misas) and archived at zenodo (https://doi.org/10.5281/zenodo.4106472). Documentation (https://chfc-cmi.github.io/misas) consists of both a description of the application programming interface (API) usage and tutorials, which include the two case studies. Continuous integration is provided by GitHub actions, where any version pushed to the master branch is tested by running all cells of each notebook in a defined minimal environment. Installable packages are released to

the python package index (https://pypi.org/) for easy installation. misas builds on top of multiple other open source projects, including fastai (24), pytorch (25), torchio (26), and numpy (27).

The software is generic and framework-independent and was tested with pytorch, fastai v1, fastai v2, and tensorflow (28). In order to apply misas to new data, images and masks can be imported into misas from a variety of sources, e.g. from png images. The model needs to provide a prediction function that takes an image and returns a predicted segmentation mask (Fig. 1). If the model requires a defined input size, an optional function for size preparation can be provided. misas can be easily extended with custom transformation functions, which require input and output as instances of the Image/ImageSegment fastai classes, but can do arbitrary operations on the data in between.

Results And Discussion

To the best of our knowledge misas is the first tool of its kind. Therefore, there is no systematic comparison and benchmarking with related tools. The following two case studies are presented in great detail in the online documentation, including source code, images and graphs. As documentation is written as executable notebooks they can even be interactively explored, without installation using Google Colab. In the next sections the case studies are only briefly summarized to demonstrate the main points.

Case Study I – Model Suitability

The first case study addressed the problem of producing initial training data for a deep learning-based cardiac cine segmentation framework with transfer learning to 7 T (29). On the one hand there is a public dataset of cardiac magnetic resonance images, the Data Science Bowl Cardiac Challenge (DSBCC) data (30). But the ground truth labels only contain end-systolic and end-diastolic left-ventricular volumes and not individual segmentation masks. On the other hand there is a published neural network for cardiac segmentation (further called ukbb_cardiac) (31) which is specifically trained for use with quite homogeneous data from the UK Biobank (32). Based on this scenario misas was applied to determine the optimal preparation of the DSBCC data to be used by ukbb_cardiac network. (30)

Initial application of the network to random images showed poor performance overall. To improve the performance the impact of image orientation was deciphered in a first step, showing that a rotation by 90° clockwise provided optimal results (Fig. 2). This is equivalent to transposing the axes and flipping left-right and can be explained by the fact that the ukbb_cardiac model usually takes input data from NIfTI format, where axes are stored differently compared to DICOM format. Next the sensitivity to image size becomes apparent as performance breaks down when using images larger than 256 pixels (Fig. 3). Further analyses show relatively low sensitivity to other kinds of transformations.

As a result a clear set of rules for data preparation to optimize prediction accuracy and performance was derived: ideally the images are rotated by 90° and scaled down to 256 pixels.

Case Study II – Model Robustness

The second case study showed how sensitivity analysis helps deep learning-based software users to evaluate a newly trained model. More precisely, a model was demanded for segmentation of the heart in transversal ultra-high field MR images to improve B_0 shimming performance (33). A model pre-trained on short-axis cine images at 7T (29) was fine-tuned with very little additional data (90 images from 4 subjects). It was investigated how quickly the segmentation performance collapses when dataset characteristics differ to those of the training set. Furthermore, it was examined which image features are used by the model to make its predictions and what kinds of intuitive or knowledge-based features are learned. An interesting insight, revealed by analysis of sensitivity to rotation is that the model tends to predict the heart on the right hand side of the image, even incorrectly so when it is rotated by 180°. Additionally, the impact of realistic MR artifacts on sensitivity was analyzed. The analysis of spike artifacts in different positions in k-space and different intensity reveals a high sensitivity (Fig. 4). Only spikes very close to the center of k-space and low intensity are tolerated, all other configurations lead to failure of segmentation.

Overall the model is quite sensitive to most transformations with only a small parameter range with stable predictions. Hence a decision on further training can now be made depending on the use case. As long as the model is used on data locally acquired with identical protocol and no artifacts, the model can be used as is. More data augmentation should be incorporated in re-training for the use on external data. In any case more data is required to further improve segmentation performance.

General Discussion and Limitations

A major advantage of the developed workflow is its applicability to any model. Access to original training data or anything happening within the blackbox is not required. The only requirement is access to the prediction function. Results of the sensitivity analysis are visualized as overlays on the image or as graphs of a metric over the parameter space. Both visualizations are readily interpretable and easy to understand. Analysis can help to guide decisions like pre-processing of data before usage with a model, or re-training the model with either more or less extensive data augmentation.

While the local interpretability of a single image could easily be analyzed in detail, the obtained information cannot always be transferred to any input image and is a limitation of the presented sensitivity analysis. An image which could be evaluated well should ideally be chosen as the starting point, otherwise unsatisfactory analysis results would be obtained. It might also not be straightforward to derive concrete steps how the robustness can be improved - or how a specific failure can be eliminated. Moreover, the developed software will not help to evaluate the impact of subtle differences introduced by bias that goes beyond simple transformations (like racial or gender differences). However, if there is a model for artificially introducing a certain kind of bias into an image, the impact of this bias could consequently be analyzed using misas.

It is important to note that sensitivity to a certain transformation is neither a bad nor a good thing per se and has to be interpreted in the context of the question at hand.

Furthermore, there is a close relationship between sensitivity analysis and data augmentation. A direct effect between amount and types of data augmentation and model sensitivity regarding the respective transformations is expected. However, sensitivity analysis is still useful for models for which the training process could not be influenced - or even no information on how it was trained could be assessed. Even for self-trained models with data augemtnation, sensitivity analysis can be used to check if a suitable amount of data augmentations was employed to reach the desired model robustness.

Broader Applicability and Future Developments

In the case studies sensitivity analysis was only performed on cardiac MR images. However, neither the method nor the library is restricted to this narrow application area. Both can be applied to other medical imaging areas e.g. pneumothorax segmentation or general imaging e.g. CamVid (34) without the need for further adaptions.

Future work will focus on enabling global interpretability by implementing a batch mode that works on multiple example images at once. Additionally the development of quantitative measures of sensitivity has high priority.

Conclusions

In this study, we demonstrate how sensitivity analysis can be used to get insights into generic segmentation model performance. It makes predictions more interpretable by expanding the context from single images to a whole range of related images with known transformations. Additionally, we present an open source python library that allows the scientific community to apply this approach to own data and models.

Availability And Requirements

- **Project name**: misas
- **Project home page**: https://github.com/chfc-cmi/misas
- **Operating system(s)**: Platform-independent
- **Programming language**: Python
- **Other requirements**: matplotlib, pytorch, fastai (v1.0.61), gif, tensorflow, altair, fastai2, pydicom, kornia, scikit-image, torchio
- **License**: MIT
- **Any restrictions to use by non-academics**: None

List Of Abbreviations

API application programming interface

DSBCC Data Science Bowl Cardiac Challenge

LV left ventricle

misas model interpretation through sensitivity analysis for segmentation

MY myocardium

RV right ventricle

Declarations

Ethics approval and consent to participate

Ethics approval of the local ethics committee at the University Hospital Würzburg has been granted under reference number 7/17-sc.

Consent for publication

All human volunteers gave their consent for publication using our institutional consent form.

Availability of data and materials

The source code is available in GitHub and zenodo https://github.com/chfc-cmi/misas, https://doi.org/10.5281/zenodo.4106472.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported by the German Ministry of Education and Research (grant number, 01EO1504). The funding body took no role in the design of the study, collection, analysis, and interpretation of data and in writing the manuscript.

Authors' contributions

MJA & LMS designed the study. MJA & LS developed the source code. MJA, MH & DL designed the case studies. MH & DL collected the data. MJA, MH & DL analyzed and interpreted the data. MJA wrote the initial draft of the manuscript. All authors read and approved the final manuscript.

Acknowledgements

This project is inspired by work from Max Woolf: https://github.com/minimaxir/optillusion-animation.

References

1. Havaei M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y, et al. Brain tumor segmentation with Deep Neural Networks. Medical Image Analysis. 2017;35:18-31.
2. Eijgelaar RS, Visser M, Müller DMJ, Barkhof F, Vrenken H, Herk Mv, et al. Robust Deep Learning–based Segmentation of Glioblastoma on Routine Clinical MRI Scans Using Sparsified Training. 2020;2(5):e190103.
3. De Fauw J, Ledsam JR, Romera-Paredes B, Nikolov S, Tomasev N, Blackwell S, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. Nat Med. 2018;24(9):1342-50.
4. Jakhar K, Bajaj R, Gupta RJA. Pneumothorax Segmentation: Deep Learning Image Segmentation to predict Pneumothorax. 2019;abs/1912.07329.
5. Chen C, Qin C, Qiu H, Tarroni G, Duan J, Bai W, et al. Deep learning for cardiac image segmentation: A review. Frontiers in Cardiovascular Medicine. 2020;7:25.
6. Leiner T, Rueckert D, Suinesiaputra A, Baeßler B, Nezafat R, Išgum I, et al. Machine learning in cardiovascular magnetic resonance: basic concepts and applications. Journal of Cardiovascular Magnetic Resonance. 2019;21(1):61.
7. Litjens G, Ciompi F, Wolterink JM, de Vos BD, Leiner T, Teuwen J, et al. State-of-the-Art Deep Learning in Cardiovascular Image Analysis. JACC: Cardiovascular Imaging. 2019;12(8, Part 1):1549-65.
8. Petersen SE, Abdulkareem M, Leiner T. Artificial Intelligence Will Transform Cardiac Imaging—Opportunities and Challenges. Frontiers in Cardiovascular Medicine. 2019;6.
9. Reyes M, Meier R, Pereira S, Silva CA, Dahlweid F-M, Tengg-Kobligk Hv, et al. On the Interpretability of Artificial Intelligence in Radiology: Challenges and Opportunities. Radiology: Artificial Intelligence. 2020;2(3):e190043.
10. Subbaswamy A, Saria S. From development to deployment: dataset shift, causality, and shift-stable models in health AI. Biostatistics. 2020;21(2):345-52.
11. Oakden-Rayner L, Dunnmon J, Carneiro G, Ré CJPotACoH, Inference,, Learning. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. 2020.
12. Kaushal A, Altman R, Langlotz C. Geographic Distribution of US Cohorts Used to Train Deep Learning Algorithms. JAMA. 2020;324(12):1212-3.
13. Geirhos R, Jacobsen J-H, Michaelis C, Zemel R, Brendel W, Bethge M, et al. Shortcut Learning in Deep Neural Networks. 2020;abs/2004.07780.
14. Guo FM, Ng M, Goubran M, Petersen SE, Piechnik SK, Neubauer S, et al. Improving cardiac MRI convolutional neural network segmentation on small training datasets and dataset shift: A continuous kernel cut approach. Medical Image Analysis. 2020;61.
15. Vilone G, Longo LJA. Explainable Artificial Intelligence: a Systematic Review. 2020;abs/2006.00093.
16. Kokhlikyan N, Miglani V, Martín M, Wang E, Alsallakh B, Reynolds J, et al. Captum: A unified and generic model interpretability library for PyTorch. 2020;abs/2009.07896.

17. Sundararajan M, Taly A, Yan Q, editors. Axiomatic Attribution for Deep Networks. ICML; 2017.
18. Selvaraju RR, Das A, Vedantam R, Cogswell M, Parikh D, Batra DJIJoCV. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. 2019;128:336-59.
19. Dhamdhere K, Sundararajan M, Yan QJA. How Important Is a Neuron? 2019;abs/1805.12233.
20. Widrow B, Hoff ME, editors. Adaptive Switching Circuits. 1960 IRE WESCON Convention Record; 1960: IRE.
21. Shu H, Zhu H. Sensitivity Analysis of Deep Neural Networks. Proceedings of the AAAI Conference on Artificial Intelligence. 2019;33:4943-50.
22. Teodoro G, Kurç TM, Taveira LFR, Melo ACMA, Gao Y, Kong J, et al. Algorithm sensitivity analysis and parameter tuning for tissue image segmentation pipelines. Bioinformatics. 2017;33(7):1064-72.
23. Knuth DE. Literate Programming. Comput J. 1984;27(2):97-111.
24. Howard J, Gugger S. fastai: A Layered API for Deep Learning. Information. 2020;11(2):108.
25. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. 2019:8024–35.
26. Pérez-García F, Sparks R, Ourselin S. TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. arXiv:200304696 [cs, eess, stat]. 2020.
27. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020;585(7825):357-62.
28. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2016;abs/1603.04467.
29. Ankenbrand MJ, Lohr D, Schlötelburg W, Reiter T, Wech T, Schreiber LM. A Deep Learning Based Cardiac Cine Segmentation Framework for Clinicians - Transfer Learning Application to 7T. medRxiv. 2020:2020.06.15.20131656.
30. Booz Allen Hamilton. Data Science Bowl Cardiac Challenge Data. https://www.kaggle.com/c/second-annual-data-science-bowl: kaggle.com; 2016.
31. Bai W, Sinclair M, Tarroni G, Oktay O, Rajchl M, Vaillant G, et al. Automated cardiovascular magnetic resonance image analysis with fully convolutional networks. Journal of Cardiovascular Magnetic Resonance. 2018;20.
32. Petersen SE, Matthews PM, Francis JM, Robson MD, Zemrak F, Boubertakh R, et al. UK Biobank's cardiovascular magnetic resonance protocol. Journal of Cardiovascular Magnetic Resonance. 2016;18(1):8.
33. Hock M, Terekhov M, Stefanescu MR, Lohr D, Herz S, Reiter T, et al. B0 shimming of the human heart at 7T. Magn Reson Med. 2020;85(1):182-96.
34. Brostow G, Fauqueur J, Cipolla RJPRL. Semantic object classes in video: A high-definition ground truth database. 2009;30:88-97.

Figures

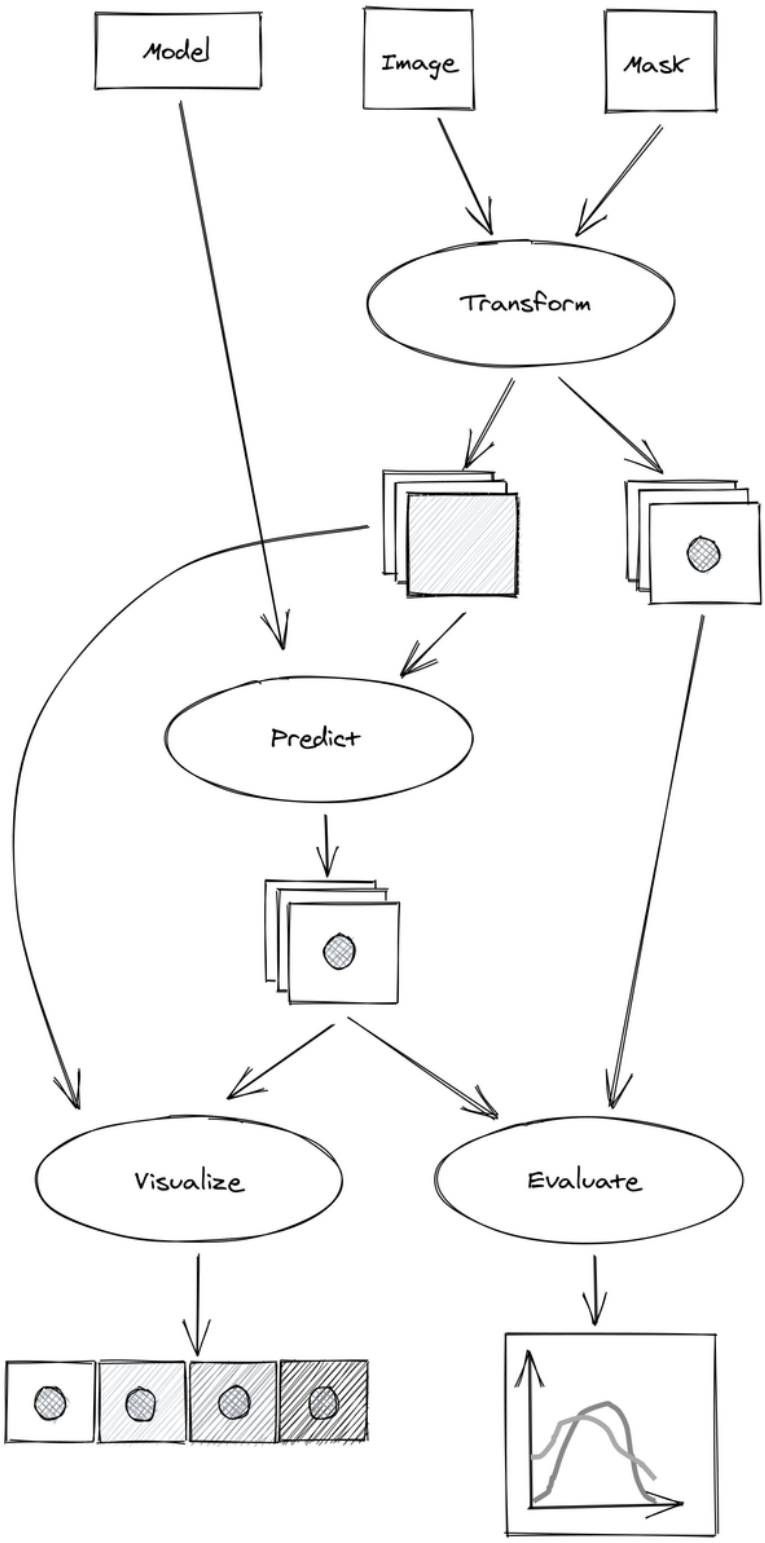

Figure 1

Schematic workflow of misas. Input are a model, an image and optionally a ground truth mask. Images are transformed (e.g. rotated, cropped, zoomed) across a parameter space. Predictions are made on these

transformed images and the result is visualized or evaluated using the masks (accordingly transformed if necessary).

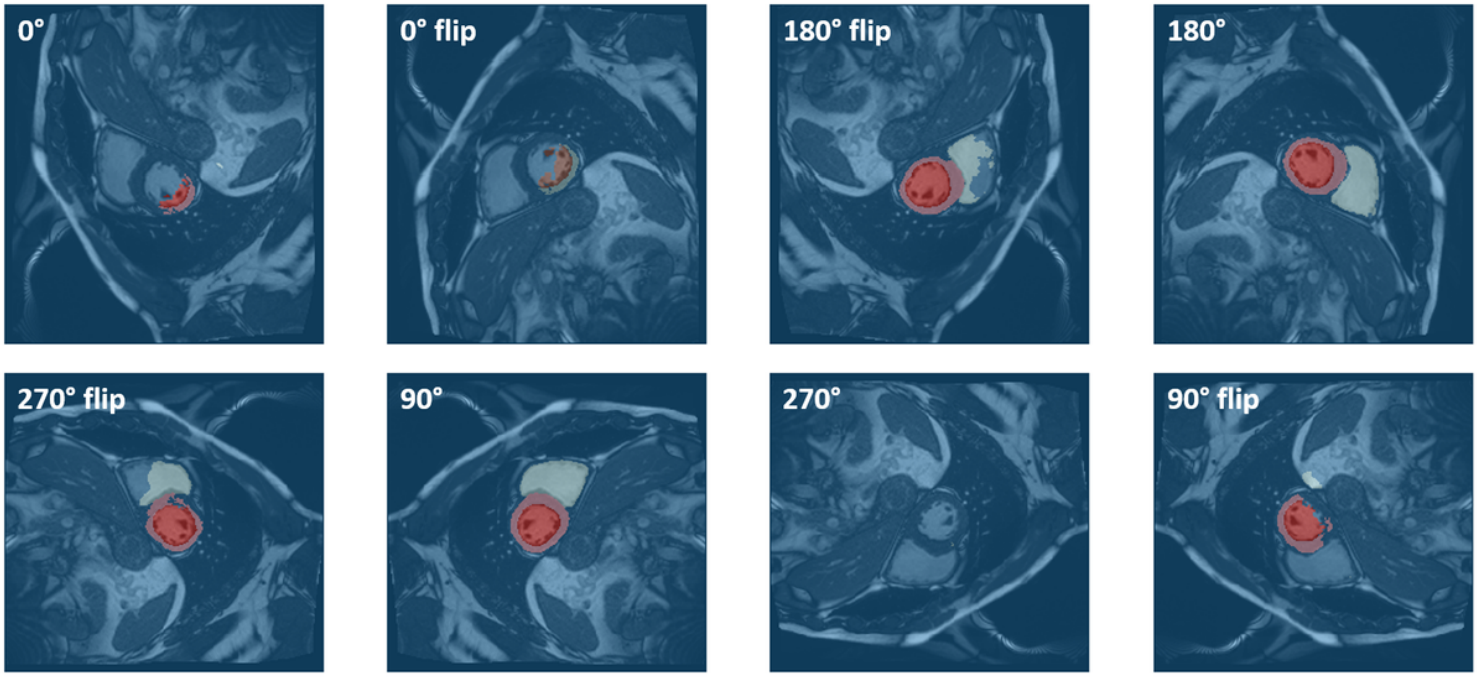

Figure 2

Segmentation result of ukbb_cardiac network (31) on an image from the Data Science Bowl Cardiac Challenge Data (30) on all possible rotations and flips. Performance is highly dependent on image orientation. Rotation angle (clockwise) and flip status (up/down) given.

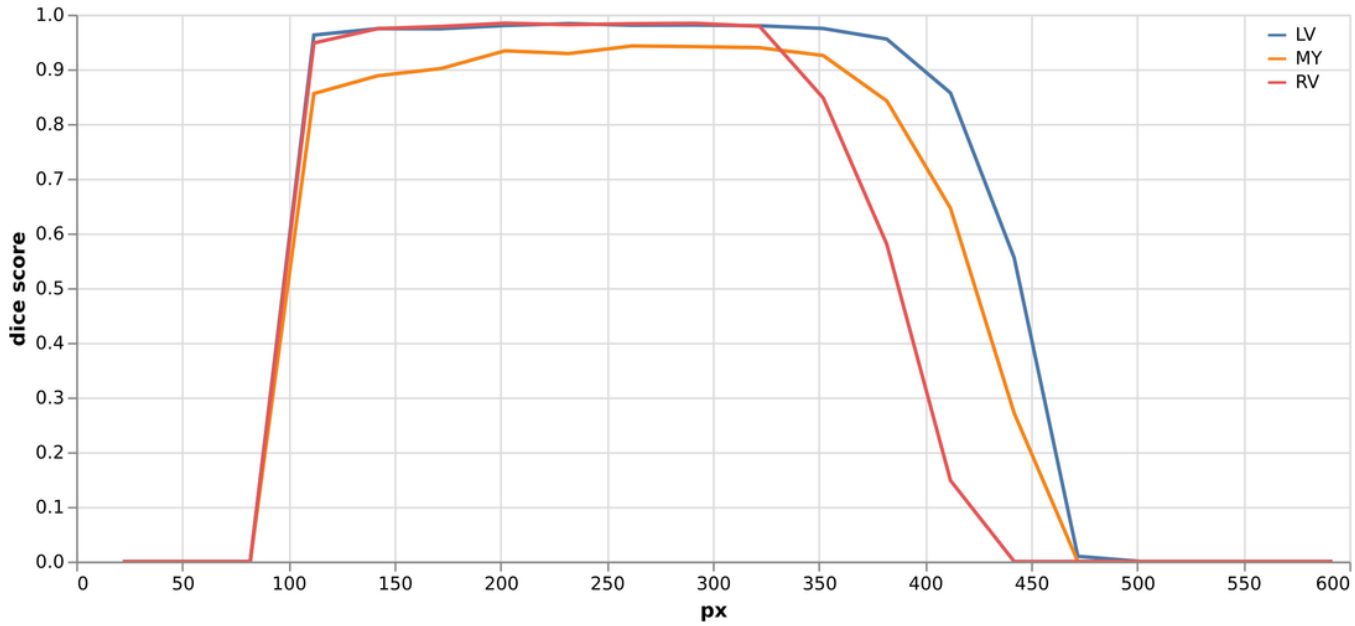

Figure 3

Dice score for each tissue (left ventricle (LV), right ventricle (RV), myocardium (MY)) depending on image size. Small images (<100px) have 0 dice for all classes, same is true for large images (>500px). There is

quite a broad range ~120-320px where predictions are stable.

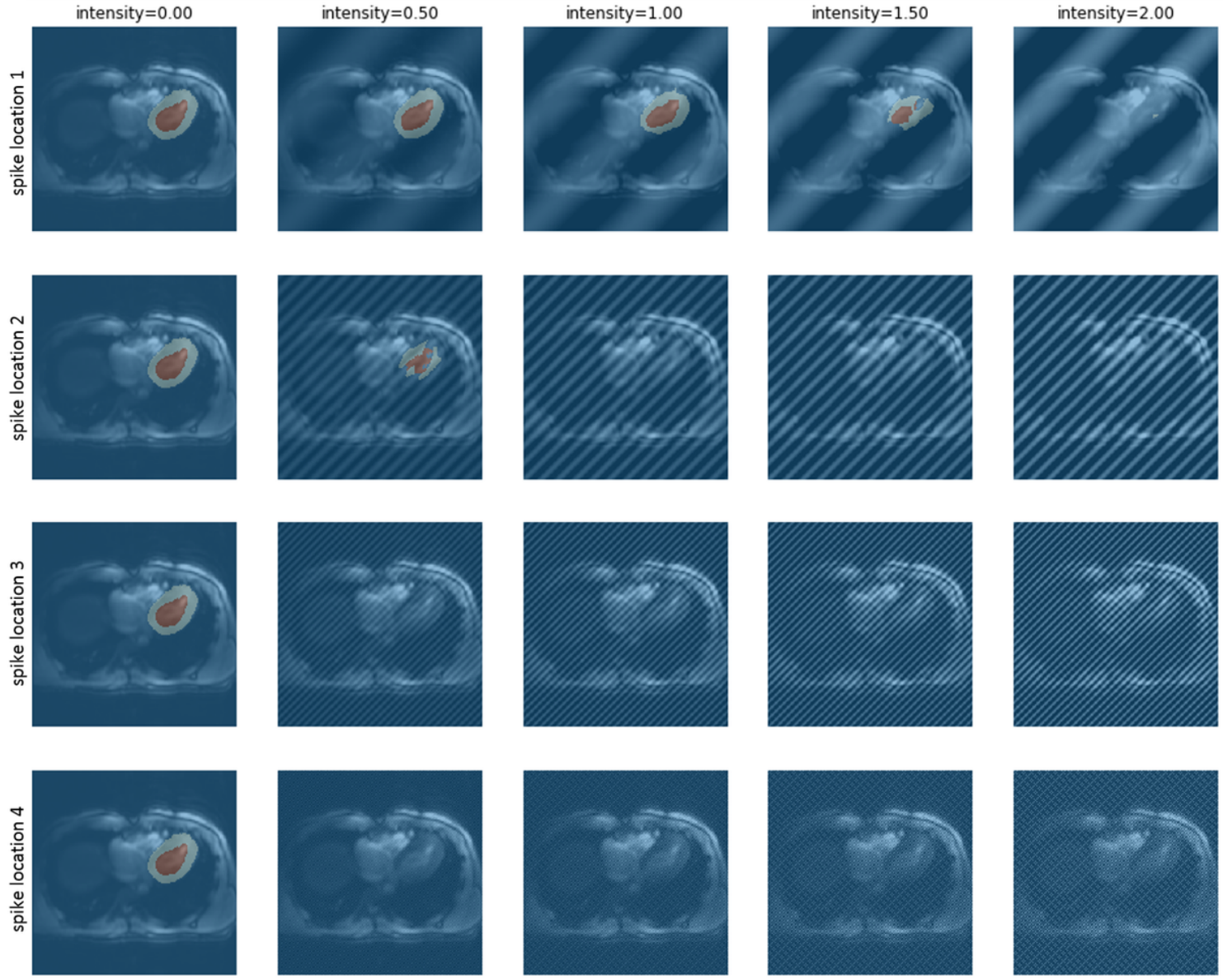

Figure 4

Segmentation performance on transversal slices with simulated spike artifacts of different localization in k-space (rows) and intensity (columns). Intensity parameter denotes the intensity of the spike relative to the original maximum intensity. From top to bottom the location in k-space moves further from the center.