UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Physics and Technology

Leveraging Supervoxels for Medical Image Volume Segmentation With Limited Supervision

Stine Hansen

A dissertation for the degree of Philosophiae Doctor

October 2022

Abstract

The majority of existing methods for machine learning-based medical image segmentation are supervised models that require large amounts of fully annotated images. These types of datasets are typically not available in the medical domain and are difficult and expensive to generate. A wide-spread use of machine learning based models for medical image segmentation therefore requires the development of data-efficient algorithms that only require limited supervision.

To address these challenges, this thesis presents new machine learning methodology for unsupervised lung tumor segmentation and few-shot learning based organ segmentation. When working in the limited supervision paradigm, exploiting the available information in the data is key. The methodology developed in this thesis leverages automatically generated supervoxels in various ways to exploit the structural information in the images.

The work on unsupervised tumor segmentation explores the opportunity of performing clustering on a population-level in order to provide the algorithm with as much information as possible. To facilitate this population-level across-patient clustering, supervoxel representations are exploited to reduce the number of samples, and thereby the computational cost.

In the work on few-shot learning-based organ segmentation, supervoxels are used to generate pseudo-labels for self-supervised training. Further, to obtain a model that is robust to the typically large and inhomogeneous background class, a novel anomaly detection-inspired classifier is proposed to ease the modelling of the background. To encourage the resulting segmentation maps to respect edges defined in the input space, a supervoxel-informed feature refinement module is proposed to refine the embedded feature vectors during inference. Finally, to improve trustworthiness, an architecture-agnostic mechanism to estimate model uncertainty in few-shot segmentation is developed.

Results demonstrate that supervoxels are versatile tools for leveraging structural information in medical data when training segmentation models with limited supervision.

Acknowledgements

At the end of this four-year PhD project, there are a number of people I would like to thank for helping me along the way.

First and foremost, I would like to thank my supervisor Professor Robert Jenssen for his guidance, support, and optimism through the last four years. It has been a pleasure to work with you and learn from you.

I also want to thank Stian Normann Anfinsen for introducing me to the machine learning group during my master's and for encouraging me to apply for this PhD position when the opportunity arose.

Further, I would like to thank my co-authors for being part of this project and sharing your knowledge with me over these years. I look forward to our continued collaboration.

To everyone in the UiT Machine Learning group, thank you for all the good discussions, your support, and for making the office a great place to be. The past four years would not have been the same without you!

I would also like to thank my committee members for taking the time to read my dissertation and attending the defense.

To my friends, thank you for putting up with me during the busy periods, I look forward to seeing you more frequently again. Finally, to my family and to Michael, thank you for being there for me throughout these years. I could not have done this without you!

Stine Hansen
Tromsø, October 2022

Contents

List of Figures

List of Abbreviations

AI Artificial Intelligence

ALPNet Adaptive Local Prototype Pooling Network

CANet Class-Agnostic Segmentation Network

CE Cross-Entropy

CHAOS Combined Healthy Abdominal Organ Segmentation

CNN Convolutional Neural Network

CT Computed Tomography

DL Deep Learning

FDA Food and Drug Administration

FDG Fluorodeoxyglucose

FSL Few-Shot Learning

GPU Graphics Processing Unit

MC Monte Carlo

ML Machine Learning

MLP Multilayer Perceptron

MRI Magnetic Resonance Imaging

MST Minimum Spanning Tree

PANet Prototype Alignment Network

PAR Prototype Alignment Regularization

PET Positron Emission Tomography

ReLU Rectified Linear Unit

SLIC Simple Linear Iterative Clustering

SUV Standard Uptake Value

VI Variational Inference

1 Introduction

Machine learning (ML) is a sub-field of artificial intelligence (AI) that concerns algorithms that learn to make predictions by exploiting patterns in collected data. In the last decade, advances in ML have revolutionized the field of computer vision, yielding human-level performance in tasks such as image classification, image segmentation, and object detection [He et al., 2015; Greenwald et al., 2022; Kazemzadeh et al., 2022]. This recent success has led to rapid adoption of ML-based solutions across a variety of fields, including the healthcare sector. A study mapping the landscape of ML-enabled medical devices that have been approved by the US Food and Drug Administration (FDA) showed that the number of devices quickly increased from a total number of 27 devices in 2015 to 343 approved devices in June 2021 [Zhu et al., 2022]. The majority (70.3 %) of these devices are radiology related, aimed at assisting the different steps in the imaging process, all the way from patient positioning during image acquisition, to image reconstruction, to diagnostics and triage assistance. One such example is the *AI-Rad Companion*, provided by Siemens. This is an AI-based decision support tool designed to help assessment during radiology examinations. Specifically, it provides a collection of algorithms that help segmenting, measuring, and highlighting relevant anatomies in computed tomography (CT), magnetic resonance imaging (MRI), and X-ray images, aiming to save the clinicians' time and increase diagnostic precision [Siemens Healthineers, 2021].

The recent ML success is often attributed to three key factors: i) advancements in deep learning (DL) methodology, a sub-field within ML that concerns training

of deep neural networks, ii) availability of more computing power via graphics processing units (GPUs), and iii) availability of large annotated data sets for training [Esteva et al., 2021]. The latter factor, however, poses a significant challenge in the widespread implementation of fully supervised DL solutions within the healthcare industry. Sufficiently large and annotated data sets are typically *not* available due to a variety of reasons. First, collecting and sharing medical datasets is challenging due to strict requirements for patient privacy [Peng and Wang, 2021]. Moreover, differences in acquisition protocols, sensor differences, and diversity in patient population further complicates the task. Second, annotating data is a resource-intensive process, especially for segmentation tasks, and is associated with tedious and time-consuming manual labour performed by experienced domain experts. As an example, training the heart segmentation device in the *AI-Rad Companion* involved delineating target structures in over 650 CT data sets [Siemens Healthineers, 2021]. Knowing that manually annotating *one* single image may take from a few minutes up to several hours, depending on the complexity of the structures [Wang et al., 2021a], the data annotation process constitutes a major roadblock in the implementation of fully supervised algorithms in the healthcare domain. Therefore, to facilitate a widespread application of ML-based solutions within healthcare, it is necessary to design methods that can generalize well with limited supervision[1].

The focus of this thesis is on the development of new ML algorithms for medical image segmentation with limited supervision. In particular, completely unsupervised solutions to lung tumor segmentation and solutions that only require a few labeled samples to perform organ segmentation (few-shot learning) are proposed. To exploit the data-specific opportunities of medical images, automatically generated supervoxels are exploited in various ways to solve key challenges related to the task. These key challenges and opportunities are discussed further in the following section.

1.1 Key Challenges and Opportunities

To efficiently learn to segment medical images with limited supervision, it is crucial to exploit the enormous pool of information that is contained in unlabeled data. However, the majority of work within computer vision has focused on fully supervised learning, leaving the use of unlabeled data underexplored. To realize the full potential of all the available data in the medical domain, new methodological advances are thus required [Peng and Wang,

1. In this thesis, *limited supervision* refers to the amount of labeled samples, and not to their quality.

2021].

In particular, prior work in computer vision has mostly concerned segmentation of natural images in 2D, whereas the medical segmentation tasks considered in this thesis involve volumetric images. This requires the development of new solutions that can leverage the 3D structure of these images efficiently.

Another challenge is related to the limited supervised training signals, requiring the development of new training mechanisms that can exploit the underlying structure in the unlabeled data and the potentially small amounts of available labeled data. This requires the exploitation of prior information about the data and task, as well as additional constraints to ensure the learning of robust models.

Further, the adaption of ML-based solutions for medical image segmentation comes with additional challenges related to the safety-critical nature of the field. For instance, medical experts need to be provided with a notion of uncertainty along with the prediction. There is thus a desire for medical segmentation approaches that can learn in the presence of limited supervision, while still providing uncertainty estimates.

While medical image segmentation with limited supervision is an inherently difficult task, the nature of the data introduces unique opportunities (in comparison to natural images) that can—and should—be exploited in designing new data-efficient models.

Firstly, because the acquisition of medical images typically follows a standardized protocol, the variations between images are relatively small compared to natural images, see examples in Figure 1.1. The variations are generally limited to anatomical variations between patients and small variations related to the acquisition process. This consistency across image volumes in a patient population can be exploited by the model to more easily find robust patterns in the data.

Further, classes of interest in common medical image segmentation tasks, such as organ segmentation, are often relatively spatially homogeneous. This presents an opportunity to exploit supervoxels within the model design, as they define homogeneous sub-regions of an image and can be generated automatically.

Figure 1.1: Examples showing variations within the dog class in the ImageNet benchmark dataset [Deng et al., 2009] (top) and variations between abdominal MRIs in the Combined Healthy Abdominal Organ Segmentation (CHAOS) dataset [Kavur et al., 2019](bottom).

1.2 Research Objectives

To address the key challenges above, this thesis proposes novel methodology for medical image segmentation with limited supervision. The main objectives of the thesis are summarized as follows:

1. Leverage data-specific opportunities in medical images through automatically generated supervoxels to train segmentation models with limited supervision.

2. Reconsider the current approach to few-shot medical image segmentation to obtain models that are robust to a large and inhomogeneous background class.

3. Design ad-hoc approaches to quantify the uncertainty in segmentation predictions and use this information to improve performance.

1.3 Proposed Approaches

The methodology developed in this thesis addresses the first research objective in three different ways: In Paper I, the supervoxels are used to reduce the computational cost of a population-level clustering approach to unsupervised

segmentation. In Paper II, supervoxels are used to generate pseudo-labels for self-supervised training of a few-shot segmentation network. In Paper III, supervoxels are used to refine features in the embedding space to obtain segmentations that respect edges and fine-grained details in the input space.

Research objective 2 is mainly addressed in Paper II, where a new approach to few-shot segmentation is presented. Whereas the foreground class in medical image segmentation tasks typically is relatively homogeneous, the background class is not. To bypass the challenge of modeling the large and inhomogeneous background class with prototypes, the proposed methodology in this thesis is inspired by the problem of anomaly detection and refrains from explicitly modelling the difficult background class. Paper III further builds on this framework by improving the inference phase of the anomaly detection-inspired approach.

The third research objective is addressed in Paper III where a Monte-Carlo dropout [Gal and Ghahramani, 2016] inspired approach to uncertainty estimation for prototypical few-shot segmentation networks is proposed. The resulting uncertainty maps are further exploited to guide a proposed feature refinement.

1.4 Brief Summary of Included Papers

The thesis' main contribution are the three included papers which are briefly summarized in the following. Figure 1.2 provides an overview of the topics considered in the various papers.

I) Stine Hansen, Samuel Kuttner, Michael Kampffmeyer, Tom-Vegard Markussen, Rune Sundset, Silje Kjærnes Øen, Live Eikenes, and Robert Jenssen, "**Unsupervised supervoxel-based lung tumor segmentation across patient scans in hybrid PET/MRI**", Expert Systems with Applications, 2021.

II) Stine Hansen, Srishti Gautam, Robert Jenssen, and Michael Kampffmeyer, "**Anomaly Detection-Inspired Few-Shot Medical Image Segmentation Through Self-Supervision With Supervoxels**", Medical Image Analysis, 2022.

III) Stine Hansen, Srishti Gautam, Suaiba Amina Salahuddin, Michael Kampffmeyer, and Robert Jenssen "**ADNet++: A few-shot learning framework for multi-class medical image volume segmentation with uncertainty-guided feature refinement**", In submission, 2022.

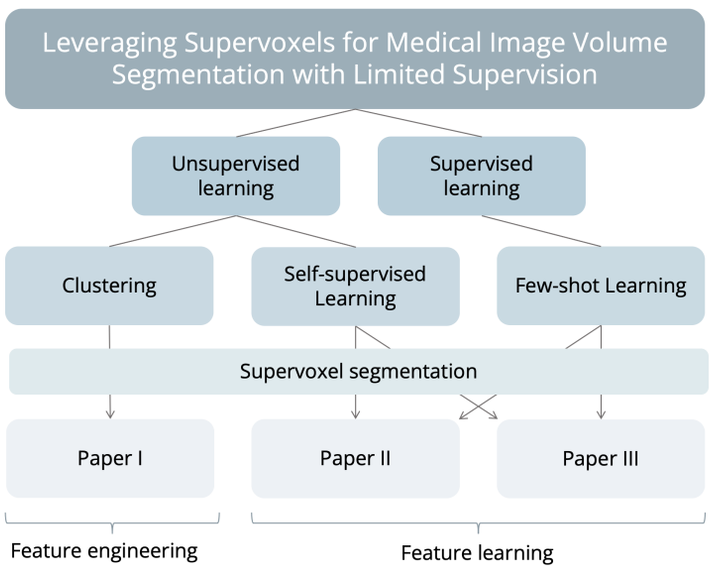


Figure 1.2: Overview of the topics that the various papers address.

Paper I This paper investigates the potential of unsupervised lung tumor segmentation from positron emission tomography (PET)/MRI through a two-step supervoxel-based clustering framework. Instead of performing computationally expensive and noise sensitive voxel-wise clustering, the proposed two-step approach works on the supervoxel-level, thereby enabling noise-robust population-level clustering.

Paper II This paper proposes an anomaly detection-inspired approach to few-shot medical image segmentation that addresses the challenge of modelling the large and inhomogeneous background class. Whereas previous approaches attempt to model the background class with one or multiple prototypes, Paper II suggests to *not* explicitly model the background, but to treat background voxels as *anomalies* from the well defined foreground class. Thus, if a voxel deviates too much, as defined by a learned threshold, from the normal (foreground) class, it should be classified as background. This results in a model that is robust to large variations in the background class. The paper further develops a self-supervision task that exploits supervoxels to train the network in an unsupervised manner.

Paper III This paper builds on the framework in Paper II by improving the inference phase to produce more accurate and more trustworthy predictions. The proposed methodology include a supervoxel-based feature refinement module that aims to produce predictions that respect edges in the input image. Further, to avoid ambiguous voxel predictions, the paper extends the few-shot medical

image segmentation to multi-class segmentation. Finally, the paper proposes a mechanism to provide uncertainty maps for the predictions and illustrates how these can be used to guide the proposed feature refinement.

1.5 Other Papers

4. Stine Hansen, Srishti Gautam, Robert Jenssen, and Michael Kampffmeyer "**Anomaly Detection-Inspired Few-Shot Medical Image Segmentation Through Self-Supervision**", Norwegian Society for Image Processing and Machine Learning Conference (NOBIM), 2021.

5. Srishti Gautam, Marina M.-C. Höhne, Stine Hansen, Robert Jenssen, and Michael Kampffmeyer, "**This looks more like that: Enhancing Self-Explaining Models by Prototypical Relevance Propagation**", Norwegian Society for Image Processing and Machine Learning Conference (NOBIM), 2021.

6. Srishti Gautam, Marina M.-C. Höhne, Stine Hansen, Robert Jenssen, and Michael Kampffmeyer. "**Demonstrating The Risk of Imbalanced Datasets in Chest X-ray Image-based Diagnostics by Prototypical Relevance Propagation**", IEEE 19th International Symposium on Biomedical Imaging (ISBI), 2022.

7. Srishti Gautam, Marina M.-C. Höhne, Stine Hansen, Robert Jenssen, and Michael Kampffmeyer. "**This looks more like that: Enhancing Self-Explaining Models by Prototypical Relevance Propagation**", In submission, 2022.

8. Suaiba Amina Salahuddin, Stine Hansen, Srishti Gautam, Michael Kampffmeyer, and Robert Jenssen. "**A self-guided anomaly detection-inspired few-shot segmentation network**", Colour and Visual Computing Symposium (CVCS), 2022.

9. Srishti Gautam, Ahcene Boubekki, Stine Hansen, Suaiba Amina Salahuddin, Robert Jenssen, Marina M.-C. Höhne, and Michael Kampffmeyer. "**ProtoVAE: A Trustworthy Self-Explainable Prototypical Variational Model**", 36th Annual Conference on Neural Information Processing Systems (NeurIPS), 2022.

1.6 Reading Guide

This thesis is structured into four parts: *I) Machine Learning Methodology*, *II) Medical Image Segmentation with Limited Supervision*, *III) Summary of Research*, and *IV) Included Papers*.

Machine Learning Methodology provides the basic machine learning concepts (Chapter 2) and deep learning theory (Chapter 3) relevant for this thesis, and introduces important concepts related to learning with limited supervision (Chapter 4). *Medical Image Segmentation From Limited Labeled Data* briefly introduces the principles of the imaging modalities relevant for Paper I-III, discusses challenges related to the data (Chapter 5), and provides a brief overview of existing approaches to solve the segmentation tasks considered in this thesis (Chapter 6). *Summary of Research* provides a summary of the three included papers, their scientific contributions, and the specific contributions of the author (Chapter 7 – 9). Finally, it provides concluding remarks of the work. *Included Papers* lists the included papers in the thesis.

Part I

Machine Learning Methodology

2

Machine Learning Basics

The focus of this thesis is on medical image segmentation through machine learning algorithms, that is, "*algorithms that improve automatically through experience*" [Mitchell, 1997]. This chapter briefly reviews the basic machine learning concepts relevant for the thesis.

2.1 Notation

A machine learning algorithm learns from the training dataset $\mathcal{D}_{tr}$ with the goal of generalizing well to new, unseen samples, represented by the test dataset $\mathcal{D}_{te}$. A dataset contains a finite number of data points, each defined by a d dimensional column vector $\boldsymbol{x} = [x_1, \ldots, x_d]^T$, where each dimension represent one measurable *feature*, or property, of the sample. Taking RGB color images as an example, the data points can correspond to the image pixels, represented by three-dimensional feature vectors that indicate the pixels' intensities in the red, green, and blue channels. A sample might also be associated with a *label* $\boldsymbol{y} = [y_1, \ldots, y_m]$, indicating the m target value(s) the machine learning algorithm is to predict. In the example with the RGB images, a task could be to classify the pixels into two classes: foreground pixels and background pixels. Each sample in the training set would then be associated with a label $y \in \{0, 1\}$ indicating the true class of that pixel.

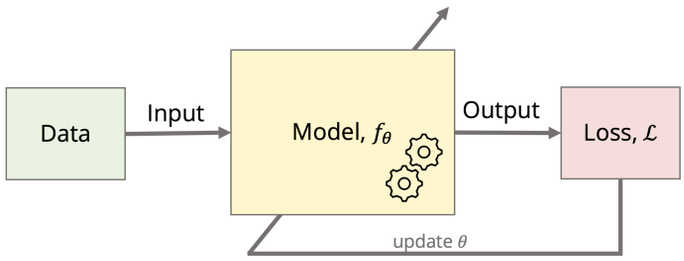


Figure 2.1: The concept of model training in machine learning. Based on the training data, the model is learned by tuning its parameters to solve a specific task.

2.2 Machine learning tasks

Depending on the available data, machine learning can be used to solve a variety of tasks, including classification, regression, ranking, clustering, detection, translation, density estimation, and, most important for this thesis, segmentation. A segmentation task consists in partitioning an input image $\boldsymbol{X}$ into n non-overlapping sub-regions $\{R_i\}_{i=1}^{n}$, such that:

$$\boldsymbol{X} = \bigcup_{i=1}^{n} R_i, \tag{2.1}$$

where $\cup$ represent the set union [Gonzalez and Woods, 2008]. This task can be solved via classification algorithms that classify each pixel into one of n predefined classes, or via clustering algorithms that cluster the pixels into n natural groups.

2.3 Learning paradigms

By "experiencing" training data, machine learning algorithms build models that can solve tasks by exploiting patterns in the data. Most algorithms[1] involve a training phase where a model $f_{\boldsymbol{\theta}}$ is learned by fitting its parameters $\boldsymbol{\theta}$ to solve a specific task by training on the training data $\mathcal{D}_{Tr}$. The training typically involves the optimization of a loss function $\mathcal{L}$, and can be solved in one step (analytical solution) or numerically in an iterative manner. The concept of iterative model training is illustrated in Figure 2.1.

1. Some algorithms are so called *lazy* learners, and do not involve a training phase, but store the training data and delay processing until a test sample is presented, like the k-nearest neighbour classifier. These algorithms require no training at the cost of a more expensive inference.

Depending on the availability of labels during training, most machine learning algorithms fall into the categories of supervised learning, unsupervised learning, or a mix between the two.

Supervised In a supervised machine learning problem, the algorithm is provided with both the input data and the desired output. The training dataset is thus given as a number of sample-and-label pairs: $\mathcal{D}_{Tr} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$, where N is the total number of training samples. The algorithms then concern the learning of a mapping from the input to the output.

Unsupervised In an unsupervised machine learning problem, the training data does not contain any labels $\mathcal{D}_{Tr} = \{\boldsymbol{x}_i\}_{i=1}^{N}$ and the algorithm aims to learn useful properties by recognizing the underlying patterns in the data. Common unsupervised algorithms include clustering algorithms, dimension reduction algorithms, and representation learning algorithms.

Semi-supervised A semi-supervised learning problem is a mix between a supervised and an unsupervised problem, where the algorithm is provided with both labeled data and unlabeled data: $\mathcal{D}_{Tr} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N} \cup \{\boldsymbol{x}_i\}_{i=1}^{M}$, where typically $M \gg N$. This setting adheres well to the real world where the amount of available data often is massive, but where labeling is expensive and therefore limited to a subset of the samples.

2.4 Generalization

The overarching goal in machine learning is to obtain a robust model that performs well on unseen data outside the training set, that is, to obtain a model that *generalizes* well. The *triple trade-off* [Dietterich, 2003] states that there is a trade-off between three factors in any learning algorithm:

i) The capacity of the learned model.

ii) The amount of training data.

iii) The generalization error on unseen data.

Typically, the generalization error on new data tends to decrease with increasing amounts of training data, as the model gets a better "grasp" of the data distribution. Increasing the model capacity, on the other hand, usually involves two phases: At first, it leads to a decrease in generalization error, e.g. going from *under-fitting*, where the model is not complex enough to express the

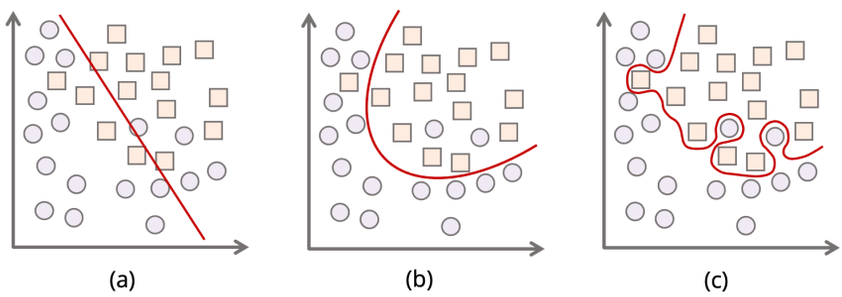


Figure 2.2: Illustration of the concepts of under-fitting and over-fitting. (a) The capacity of the model is insufficient to explain the data and the model under-fits. (b) There is a balance between model capacity and data complexity. (c) The model capacity is too high, resulting in it over-fitting to the training data, which hurts the generalization to new unseen data.

data, towards generalizing well. However, at some point, the model gets too complex and starts to *over-fit* to the training data, which leads to an increase in the generalization error. Figure 2.2 illustrates concepts of under-fitting and over-fitting to the training data in a two-class classification problem.

To obtain good generalization and avoid over-fitting to the training data, models are often *regularized* to decrease their capacity. Some common techniques are discussed in Section 3.1.3.

2.5 Feature extraction

Raw data often comes in a format that can not directly be processed by a machine learning algorithm. An important step in any machine learning framework is therefore feature extraction, transforming raw data into features that the model can handle. Nevertheless, for an algorithm to be successful, the features representing the data samples must also be relevant for the task at hand. Extracting good features is typically data-dependent *and* task-dependent, and is a challenging step that often requires expert domain knowledge.

Extracting robust features from *unstructured* data, such as images and text, is particularly difficult. Going back to the example with the RGB image, if the task is to classify the images instead of the pixels, an image of a dog should be classified as "dog"-class independently of the dog's pose, location, scale, color, and so on. Robust features should therefore be invariant to these irrelevant variations, and simply vectorizing the images is usually not sufficient. Hand-crafted features, such as *Scale Invariant Feature Transform* (SIFT) fea-

tures [Lowe, 1999], and *Histogram of Oriented Gradients* (HOG) features [Dalal and Triggs, 2005] provide image representations that are (partially) invariant to scale and rotation, and robust to noise and changes in illumination. These features have demonstrated promising performance on simple tasks, but typically fall short when the image scenes become more complex [Vondrick et al., 2013]. The following chapter considers the field of deep learning, which aims to *learn* task-specific features directly from raw data, as part of the training.

3

Deep Learning

Deep learning is a sub-field of machine learning that concerns end-to-end training of deep neural networks from raw data [LeCun et al., 2015]. In contrast to traditional machine learning models, requiring careful feature engineering, deep learning models learn to extract discriminative and relevant features directly from the raw input data. As the data propagates through the network's layers, its representation is progressively refined and tuned towards solving the task at hand. This has made deep learning a successful approach to solve different tasks in various fields. Figure 3.1 illustrates the difference between deep learning and traditional machine learning.

This chapter provides a brief introduction to the basic deep learning theory that builds the foundation of Papers II and III.

3.1 Multilayer perceptrons

Multilayer perceptrons (MLPs) are the fundamental models in deep learning. Formally, an MLP is a neural network $f_{\boldsymbol{\theta}}$, defined up to some parameters $\boldsymbol{\theta}$, and learns a mapping from the input data $\boldsymbol{x}$ to an output $\hat{\boldsymbol{y}} = f_{\boldsymbol{\theta}}(\boldsymbol{x})$. The learning refers to the adjusting of the parameters $\boldsymbol{\theta}$ to minimize some loss function $\mathcal{L}$ quantifying the disagreement between the estimated output and the desired output $\boldsymbol{y}$: $\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y})$.

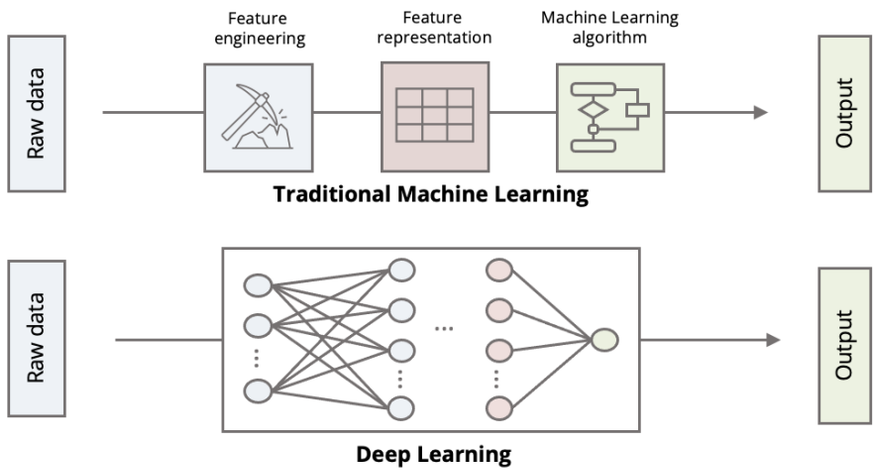


Figure 3.1: Feature engineering vs. feature learning. Traditional machine learning (top) typically involves careful feature engineering to extract features from data whereas deep learning models (bottom) learn relevant features directly from the data.

3.1.1 Architecture

The basic building block of the MLP is a simple linear combination between the input $\boldsymbol{x} \in \mathbb{R}^{d_{in}}$ and a set of weights $\boldsymbol{W} \in \mathbb{R}^{d_{in} \times d_{out}}$ (and bias terms $\boldsymbol{b} \in \mathbb{R}^{d_{out}}$), mapping the input from $\mathbb{R}^{d_{in}}$ to $\mathbb{R}^{d_{out}}$, followed by an activation function $g : \mathbb{R}^{d_{out}} \rightarrow \mathbb{R}^{d_{out}}$:

$$f_{\boldsymbol{W},\boldsymbol{b}}(\boldsymbol{x}) = g(\boldsymbol{x}^T\boldsymbol{W} + \boldsymbol{b}). \tag{3.1}$$

The MLP combines multiple such simple transformations in a chain to be able to approximate functions that have the capacity of mapping complex input data to the the desired output:

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \left(f_{\boldsymbol{W}^{(L)},\boldsymbol{b}^{(L)}} \circ \ldots \circ f_{\boldsymbol{W}^{(1)},\boldsymbol{b}^{(1)}}\right)(\boldsymbol{x}), \tag{3.2}$$

where $\boldsymbol{\theta} = [\boldsymbol{W}^{(1)}, \boldsymbol{b}^{(1)}, ..., \boldsymbol{W}^{(L)}, \boldsymbol{b}^{(L)}]$. Each function in this chain comprises one *layer* in the MLP, and the number of layers define the MLP's *depth*. The activation function g is used to introduce non-linearity in the model, and can take many forms [Goodfellow et al., 2016]. The most common activation function in deep MLPs is the rectified linear unit (ReLU), computed as the element-wise maximum of 0 and the input:

$$g_{\text{ReLU}}(\cdot) = \max(0, \cdot), \tag{3.3}$$

and is preferred over s-shaped activation functions, such as the *logistic sigmoid* function and the *tanh* activation function due to its improved gradient-flow [Glorot et al., 2011]. The activation function in the output layer is task-dependent,

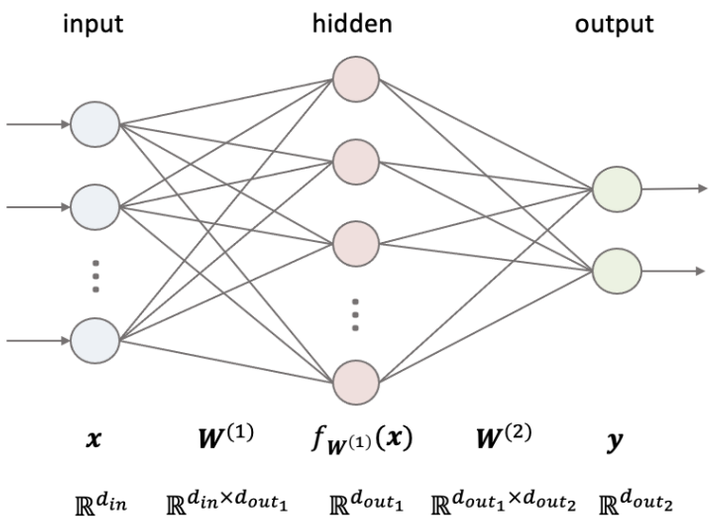


Figure 3.2: Illustration of a two-layer MLP with two-dimensional output. Note: The bias-terms have been omitted for simplicity.

and in the case of a classification objective, the scores are typically normalized with a softmax function. A simple illustration of a two-layer MLP is provided in Figure 3.2.

The general architecture of the network is partially decided by the input data and the task: In the first layer (input layer), the dimensionality d_{in} must match the dimensionality of the input d, and in the final layer (output layer), the dimensionality of the output d_{out} must match the number of classes in the case of a classification objective, or the number of response variables in a regression problem. The MLP's depth (L) and the dimensionalities of the hidden layers (layers 2 to $L-1$) are hyper-parameters of the model and can be adjusted to make the model more or less flexible.

3.1.2 Optimization

The optimization of the MLP is determined by i) the loss function, and ii) the optimization algorithm and its hyper-parameters. The loss function that is most commonly used for the classification task, is the *cross-entropy (CE)* loss function. For one prediction-label pair, this loss is defined as:

$$\mathcal{L}_{CE}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y}\right) = -\boldsymbol{y}^T \log f_{\boldsymbol{\theta}}(\boldsymbol{x}), \tag{3.4}$$

where $\log(\cdot)$ denotes the element-wise logarithm. Typically, this loss is computed as an average over a mini-batch consisting of m random samples from

the training set $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^m$ to form the total cost:

$$J(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}_{CE} \left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \boldsymbol{y}_i \right), \tag{3.5}$$

which is minimized to find the optimal weights $\boldsymbol{\theta}^*$. The optimization of the cost function is performed numerically in an iterative manner via back-propagation and (typically) a variation of *gradient descent*. The core idea in gradient descent is to compute the gradients of the cost function with respect to the weights $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$ in order to update the weights in the opposite direction of the gradients:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}), \tag{3.6}$$

where η is a hyper-parameter known as the *learning rate* and decides the step length in the direction of the negative gradient. In this vanilla gradient descent algorithm, η is typically set to a small constant and decays over the training iterations [Goodfellow et al., 2016]. Alternative algorithms, such as AdaGrad [Duchi et al., 2011] and Adam [Kingma and Ba, 2014] perform gradient descent with adaptive learning rates for all model parameters, thereby allowing larger steps to be taken along directions in the parameter space that the cost is less sensitive to.

3.1.3 Regularization

If the MLP's capacity is large enough compared to the dataset, the model can easily over-fit to the data and lose its generalization ability. This problem is exacerbated in deep MLPs that typically have no problem memorizing datasets consisting of millions of data points [Zhang et al., 2021b]. To alleviate the problem of over-fitting, it is therefore common to regularize the model. The regularization techniques come in many different forms, from early stopping, trying to stop the model training before over-fitting occur, to weight decay, penalizing the norm of the weights. This section will not try to cover all regularization techniques, but briefly visits a few common approaches. Please see [Goodfellow et al., 2016] for a comprehensive overview of regularization in deep learning.

Dropout

Dropout [Srivastava et al., 2014] provides a conceptually simple but highly effective way of regularizing neural networks. By dropping activations in the network during training, dropout can be thought of as training multiple sub-networks with shared weights. I.e. in each training iteration, a random

sub-network is sampled and trained. When dropout is added to a layer, the activations (output from the layer's activation function) $\boldsymbol{a}^{(i)} = f_{\boldsymbol{W}^{(i)},\boldsymbol{b}^{(i)}}(\boldsymbol{a}^{(i-1)}) \in \mathbb{R}^{d_{out}^{(i)}}$ are sampled with a probability p:

$$\boldsymbol{a}^{(i)} \leftarrow \boldsymbol{a}^{(i)} \odot \boldsymbol{r}, \tag{3.7}$$

where $\odot$ is the Hadamard product and the elements of $\boldsymbol{r} \in \mathbb{R}^{d_{out}^{(i)}}$ are sampled from a Bernoulli(p) distribution. During test time, instead of averaging the predictions from all the sub-networks, an approximate average is computed using the full network but with down-scaled activations, such that the output at test time remains the same as the expected output during training:

$$\boldsymbol{a}^{(i)} \leftarrow p\boldsymbol{a}^{(i)}. \tag{3.8}$$

By randomly dropping activations in this way, dropout prevents the weights in the network from building too strong dependencies that do not generalize to the test data.

Batch Normalization

Batch normalization [Ioffe and Szegedy, 2015] is an adaptive normalization of the activations in each layer of the network, primarily designed to improve optimization, but that also has a regularizing effect. Batch normalization aims to first normalize each dimension i in a layers input $\boldsymbol{x} \in \mathbb{R}^{d_{in}}$ as:

$$\hat{x}_i = \frac{x_i - \mathrm{E}[x_i]}{\sqrt{\mathrm{Var}[x_i]}}, \tag{3.9}$$

for $i = 1, ..., d_{in}$. Then, to avoid normalizing activations when it is not beneficial, the network *learns* a second transformation that has the potential to cancel out the normalization:

$$y_i = \lambda_i \hat{x}_i + \beta_i, \tag{3.10}$$

where λ_i and β_i are learned parameters.

During training, $\mathrm{E}[x_i]$ and $\mathrm{Var}[x_i]$ are estimated as the mini-batch sample mean and variance, respectively, and the generalization effect of the batch normalization comes from the noise injected to the activations by these estimates: As the estimates are dependent on the samples in the current mini-batch, samples are affected differently each time they appear in a new mini-batch.

To make the predictions deterministic during test time, only depending on the input, a running average over the mini-batch statistics computed during training is used instead.

Data augmentation

Data augmentation is a simple regularization technique that increases the dataset size by generating perturbed copies of the training samples [Goodfellow et al., 2016]. Depending on the nature of the data, different transformations may be applied to the samples in order to simulate slight variations without changing the label. This encourages model-invariance to the type of simulated variations, yielding more robust models. Common perturbation techniques in computer vision include geometric transformations (e.g. rotation, flipping, shearing, scaling, and random elastic deformation) and intensity transformations (e.g. color jittering, blurring, color inversion, and gamma corrections) [Shorten and Khoshgoftaar, 2019].

3.2 Convolutional Neural Networks

For an MLP to process an image, it would either have to i) consider each pixel individually, completely losing all spatial relations, or ii) vectorize the entire image, treating it as one long feature vector, thereby retaining some spatial relations, but limiting the applicability to small sized images. The convolutional neural networks (CNNs) build on the same principles as the MLPs, but are designed specifically to handle grid-like structured input data, such as images. Instead of of dense layers, connecting all input activations with all output activations, the CNN employs convolutional layers where the input is *filtered* by learnable convolutional filters that share weights across the image.

3.2.1 Convolution layer

The convolution operation is extensively used in traditional image processing to filter images with predefined filters in order to, for instance, blur, sharpen, or detect edge pixels [Gonzalez and Woods, 2008]. The discrete convolution between an input $\boldsymbol{X} \in \mathbb{R}^{H \times W \times d}$ and a filter $\boldsymbol{K} \in \mathbb{R}^{H' \times W' \times d'}$ is defined as:

$$(\boldsymbol{X} * \boldsymbol{K})_{i,j,k} = \sum_{h=-a}^{a} \sum_{w=-b}^{b} \sum_{d=-c}^{c} X_{i+h,j+w,k+d} K_{h,w,d}, \tag{3.11}$$

for $i = 1, \ldots, H - H' + 1$, $j = 1, \ldots, W - W' + 1$, and $k = 1, ..., d - d' + 1$, where $a = \frac{H'-1}{2}$, $b = \frac{W'-1}{2}$, and $c = \frac{d'-1}{2}$. That is, a set of shared filter weights slides across the input to compute an output value at each position. In a CNN, the convolution layer consists of a *set* of convolutions between the input to the layer $\boldsymbol{A}^{(i-1)} \in \mathbb{R}^{H \times W \times d_{in}}$ and a set of d_{out} learnable weights $\boldsymbol{W}^{(i)} \in \mathbb{R}^{H' \times W' \times d_{in}}$

(plus a bias term $b^{(i)}$), followed by an activation function g:

$$\boldsymbol{A}^{(i)} = \Big\Vert_{q=1}^{d_{out}} g\left(b_q^{(i)} + (\boldsymbol{A}^{(i-1)} * \boldsymbol{W}_q^{(i)})\right), \tag{3.12}$$

where $\Vert$ indicates the concatenation operation. The *filter size*, *stride length*, and *cardinality* are hyper-parameters of the convolution layer. The filter size is typically given by an odd integer, indicating the height and width ($H' = W'$), whereas the depth is given by the dimension of the input to the layer $d' = d_{in}$. The stride length is typically set to one, moving the filter one pixel at a time, but can be increased to spatially down-sample the input. Finally, the cardinality controls the number of groups in a channel-wise separable convolution. It is set to *one* for most networks, yielding one group of convolutions as defined above, but can be increased to reduce the number of weights in a layer: When the cardinality is set to n, the input is split into n equal-sized groups along the channel dimension, and n filters of depth d_{in}/n are used to produce n outputs of size d_{out}/n that are concatenated to produce the final output. Splitting a convolution that maps the input from d_{in} to d_{out} into n groups reduces the number of parameters from $d_{in} \times H' \times W' \times d_{out}$ to $n(\frac{d_{in}}{n} \times H' \times W' \times \frac{d_{out}}{n})$, effectively reducing the number of parameters by $1/n$. Note that both d_{in} and d_{out} must be divisible by the cardinality.

3.2.2 Pooling layer

In order to make the input's feature representation invariant to small translations, and to provide additional context for the filters by increasing their receptive fields[1], convolution layers are often followed by pooling layers in the network [Goodfellow et al., 2016]. The pooling operation reduces neighborhoods by summarizing them with one number, thereby spatially down-sampling the input. The most common pooling operation is the *max pooling* operation, which only keeps the maximum value within each neighborhood. A pooling filter of size 2×2 moving with a stride size of 2, thus reduces the spatial dimensions of the input by 50%.

3.2.3 Architectures

A CNN architecture typically consists of a contracting encoder, defined by blocks of convolution layers followed by pooling layers, and potentially (depending on the task), a small MLP or an expanding decoder. Stacking convolution layers in this manner allows the model to learn a hierarchical representation of the

1. The region in the input "seen" by each filter

input: In the initial layers, filters typically pick up on edges and corners. The next layers learn to combine these into basic shapes, which, in the deeper layers of the network, are combined into increasingly abstract levels of representation. This section provides a brief description of the two CNN architectures used as backbones in Papers II and III.

ResNet

The residual networks (ResNets) [He et al., 2016] are a family of CNNs that adopts residual connections between layers to enable efficient training of deep networks. The residual connections are identity short-cut connections that let the gradients pass directly by the layers, alleviating the problem of vanishing gradients. The ResNet-101, in particular, consists of 101 layers distributed on four types of blocks, an input convolution layer and a fully connected output layer. After each convolution, batch normalization is employed and the network has in total 42.5M learnable parameters.

ResNeXt

The ResNeXt [Xie et al., 2017] family builds on the ResNet by modifying its blocks to include convolutions with cardinalities greater that one. Specifically, the grouped convolutions are defined by the number of groups (cardinality) C and bottleneck dimension d. Thus, the ResNet is a special case of the ResNeXt with $C = 1$ and $d = 64$. A typical configuration of ResNeXt is *ResNeXt(*$32 \times 4d$*)*, e.g. a ResNeXt with cardinality $C = 4$ and bottleneck dimension $d = 4$. This network has shown improved image classification accuracy compared to ResNet, with similar number of parameters [Xie et al., 2017].

The architectures of ResNet-101 and ResNeXt($32 \times 4d$) are compared in Figure 3.3. Both ResNeXt and ResNet can be trivially extended to 3D by substituting their 2D convolutions with 3D convolutions [Hara et al., 2018]. Keeping the same filter sizes and feature dimensionalitites as in the original 2D networks, the ResNeXt scales better compared to the ResNet, with respect to the number of parameters. For instance, the 3D ResNext-101($32 \times 4d$) is a more resource efficient network than 3D ResNet-101, with close to half the number of parameters.

3.2.4 Deep image segmentation

Image segmentation refers to the process of dividing an image into its constituent regions, e.g. assigning a label to each pixel/voxel in the image. In deep

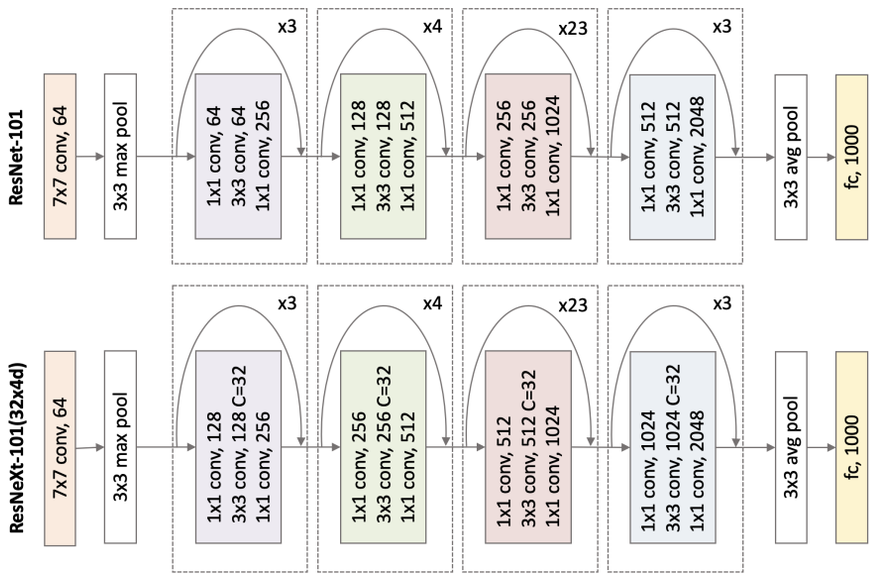


Figure 3.3: The architectures of ResNet-101 and ResNeXt-101(32x4d). The curved arrows represent the residual connections. Conv= convolutional layer, avg pool = average pooing, fc=fully connected layer.

learning, CNNs can be used in different ways when solving this difficult task and this section presents two general approaches to CNN-based deep image segmentation.

Encoder + decoder Encoder-Decoder architectures represent the most commonly used structures in deep image segmentation. These models consist of an encoder, compressing the image information into abstract high-level features, and a decoder, producing the segmentation output based on the high-level features, by slowly up-sampling the features back to the full input resolution. To alleviate the loss of spatial information caused by pooling and/or strided convolutions during encoding, popular models such as the U-Net [Ronneberger et al., 2015] make use of skip-connections that fuse high-resolution features from the encoder with up-sampled features in the decoder.

Encoder + classifier + up-sampling A simpler alternative to CNN-based image segmentation can be obtained by performing the segmentation directly in the embedding space. These models consist of a contracting encoder, like the encoder-decoder models, but do not have a symmetric decoder. Instead, they classify the abstract high-level features directly in the embedding space and up-sample the resulting segmentation mask using, for instance, bi-linear up-sampling [Chen et al., 2017a].

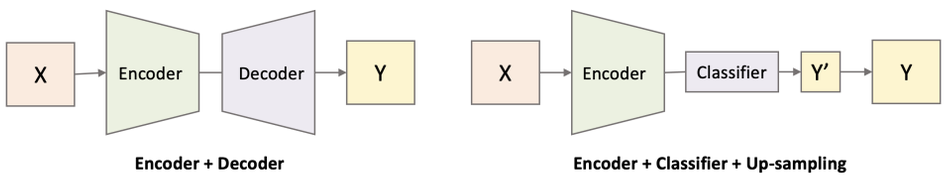


Figure 3.4: Two general approaches to CNN-based segmentation. Left: An encoder + decoder architecture that outputs the full-sized segmentation mask. Right: An encoder + decoder + up-sampling approach where the segmentation is performed on the compressed feature map in the embedding space and up-sampled to produce the final output.

While the encoder-decoder models tend to be more accurate, up-sampling in the form of a decoder typically involves learning a high number of additional weights. This can be problematic in low-data regimes where over-fitting is a challenging problem. Figure 3.4 illustrates the conceptual difference between *encoder + decoder* models and *encoder + classifier + up-sampling* models.

3.3 Predictive Uncertainty

Given an input, the models described in this chapter will *always* produce a prediction. Models, however, are not perfect and make mistakes. To be able to build systems that a user can trust, it is therefore important to quantify the model's predictive uncertainty, e.g. the model's confidence in a prediction. A *good* predictive uncertainty estimate should quantify how much the given prediction can be trusted, meaning that samples with low predictive uncertainty should be more likely to have a correct prediction.

Uncertainty estimation has become a large field of research within deep learning and the methods can be divided into two groups, based on the statistical theory that they build on: frequentist approaches and Bayesian techniques. This section focuses on Bayesian techniques, please refer to [Gawlikowski et al., 2021] for a complete review of uncertainty estimation in deep learning.

3.3.1 Bayesian Neural Networks

In a Bayesian perspective, training a model refers to discovering the parameters $\boldsymbol{\theta}$ that are likely to generate the output, given the data. Instead of estimating point estimates for $\boldsymbol{\theta}$, the aim is to compute the conditional distribution of the parameters, given the training data, the *posterior* distribution $p(\boldsymbol{\theta}|\mathcal{X}_{tr}, \mathcal{Y}_{tr})$ [Gal, 2016]. From Bayes' theorem, the posterior can be computed

as:

$$p(\boldsymbol{\theta}|\mathcal{X}_{tr}, \mathcal{Y}_{tr}) = \frac{p(\mathcal{Y}_{tr}|\mathcal{X}_{tr}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{Y}_{tr}|\mathcal{X}_{tr})}, \tag{3.13}$$

where $p(\mathcal{Y}_{tr}|\mathcal{X}_{tr}, \boldsymbol{\theta})$ is the *likelihood* of the labels given the data and some parameter setting, and $p(\boldsymbol{\theta})$ is the *prior* distribution over the parameters. Knowing the posterior distribution allows for a probabilistic prediction of test samples [Gal, 2016]:

$$p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \mathcal{X}_{tr}, \mathcal{Y}_{tr}) = \int_{\boldsymbol{\theta}'} p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \boldsymbol{\theta}')p(\boldsymbol{\theta}'|\mathcal{X}_{tr}, \mathcal{Y}_{tr})d\boldsymbol{\theta}', \tag{3.14}$$

instead of a simple point estimate. From this predictive distribution, the uncertainty can be quantified by computing its variance. However, the posterior distribution does typically *not* have an analytical solution, and numerical approximation is only feasible for very small networks [Neal, 2012].

3.3.2 Monte Carlo dropout

An alternative and scalable approach to approximate the posterior is through variational inference (VI). Instead of evaluating the posterior, an approximate posterior $q(\boldsymbol{\theta})$ is defined:

$$p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \mathcal{X}_{tr}, \mathcal{Y}_{tr}) \overset{VI}{\approx} \int_{\boldsymbol{\theta}'} p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \boldsymbol{\theta}')q(\boldsymbol{\theta}')d\boldsymbol{\theta}'. \tag{3.15}$$

In Monte Carlo (MC) dropout [Gal and Ghahramani, 2016], the approximate posterior $q(\boldsymbol{\theta})$ is a distribution over the weight matrices in each layer[2], given by:

$$\begin{aligned} \boldsymbol{W}^{(i)} &= \boldsymbol{M}^{(i)}\text{diag}(\boldsymbol{z}_i), \\ \boldsymbol{z}_i &\in \mathbb{R}^{d_{out}}, \\ z_{i,j} &\sim \text{Bernoulli}(p_i), \end{aligned} \tag{3.16}$$

for $i = 1, ..., L$, where $\boldsymbol{M}_i$ are the learnable parameters. Monte Carlo integration is then leveraged to integrate over the likelihood to estimate the predictive distribution:

$$p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \mathcal{X}_{tr}, \mathcal{Y}_{tr}) \overset{MC}{\approx} \frac{1}{T}\sum_{t=1}^{T} p(\boldsymbol{y}_{te}|\boldsymbol{x}_{te}, \boldsymbol{\theta}_t), \tag{3.17}$$

where $\boldsymbol{\theta}_t \sim q(\boldsymbol{\theta})$ [Gal, 2016]. In practice, this estimated predictive distribution is found by running T forward-passes for a test sample $\boldsymbol{x}_{te}$ through a network

2. The bias terms $\boldsymbol{b}_i$ are for simplicity typically estimated as point estimates.

with enabled dropout layers, as illustrated in Figure 3.5. To quantify the uncertainty from this estimate, metrics such as the *variation ratio*, *mutual information* or *predictive entropy* can be computed [Gal, 2016].

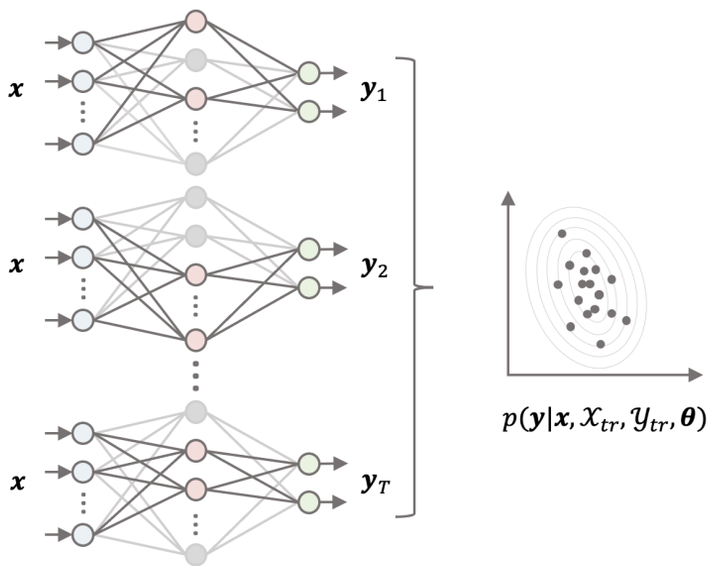


Figure 3.5: Illustration of the Monte Carlo dropout technique to estimate the predictive distribution from which the predictive uncertainty can be quantified.

4

Learning with Limited Supervision

The success of machine learning systems often rely on abundant labeled data. However, the process of collecting labels is time-consuming and expensive, making the availability of such datasets limited. Wide-spread application of machine learning models therefore depends on the development of models that can learn and generalize from data with limited supervision.

Towards this aim, various research areas have emerged. Figure 4.1 groups some of the key directions, with directions relevant for this thesis being highlighted in bold. This chapter provides an introduction to three of these: Clustering, few-shot learning, and self-supervised learning[1].

4.1 Clustering

Clustering, or cluster analysis, represents an exploratory sub-discipline of machine learning where the algorithms aim to discover the *natural* groups in the data, without relying on labels. Different clustering algorithms impose different structures on the data [Jain, 2010] which, in combination with algorithm-

1. Data augmentation is addressed in Section 3.1.3.

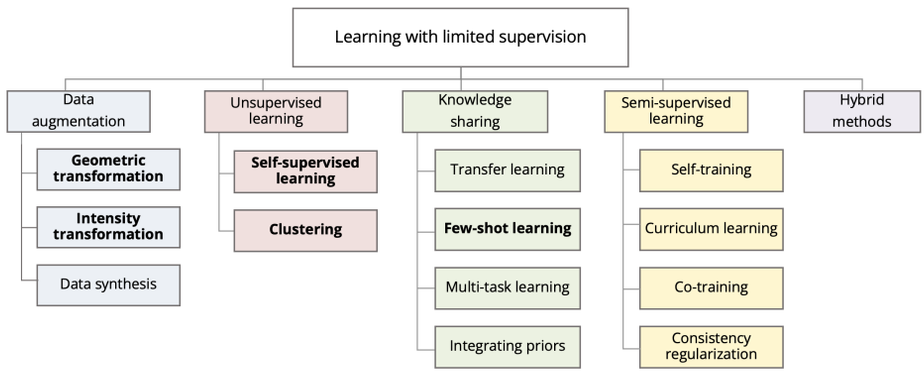


Figure 4.1: Illustration of key research areas within learning with limited supervision. Directions relevant for this thesis are highlighted in bold.

specific assumptions and hyper-parameter settings, might result in very different clustering results. This section covers the basics of the clustering algorithms explored in Paper I: k-means clustering, spectral clustering, and hierarchical clustering.

4.1.1 k-means clustering

The k-means algorithm is a two-step iterative clustering algorithm that divides the data into a predefined k number of disjoint clusters, each defined by a cluster center. Given a set of initial cluster centers, the centers are updated in order to minimize the total within-cluster variance. The following two steps are repeated until convergence [Hastie et al., 2009]:

1. Assign each sample to the cluster defined by the closest cluster center.

2. Update each cluster center with the mean of the samples assigned to its cluster.

Due to its simplicity and scalability to large sample sizes, the k-means algorithm is one of the most widely used clustering algorithms [Jain, 2010].

4.1.2 Spectral clustering

Spectral clustering can be viewed as a generalization of traditional clustering algorithms to work well when clusters are non-convex [Hastie et al., 2009]. The idea is to exploit the spectrum of the sample affinity matrix to find a low-dimensional embedding of the data, where clusters are more "obvious"

and the traditional algorithms (e.g. k-means) work better [Bengio et al., 2003]. Given an affinity matrix computed from the data, the two general steps in spectral clustering are as follows:

1. Compute a graph Laplacian from the affinity matrix, and find the m eigenvectors corresponding to its m smallest eigenvalues. Store these as the columns in a matrix F.

2. Use a traditional algorithm to perform clustering on the rows of F.

Different choices related to the computation of the affinity matrix, the graph Laplacian, and the final clustering step, give rise to algorithms with different properties, making spectral clustering a flexible framework that is applicable to a large variation of problems.

4.1.3 Hierarchical clustering

In hierarchical clustering, the algorithm produces a hierarchical representation of clusters with different solutions at each level, from one big cluster at the highest level to one cluster per sample at the lowest level [Hastie et al., 2009]. Depending on the starting point (top level or bottom level), clusters are merged or split based on a measure of proximity between pairs of clusters, resulting in a new level in the hierarchy. As opposed to k-means clustering, hierarchical clustering does not require a predefined number of desired clusters to run. The choice of clustering solution (level in the hierarchy) can be determined by manual inspection, by desired number of clusters, or automatically via a gap statistic [Tibshirani et al., 2001].

4.1.4 Superpixel clustering

Superpixel clustering is a type of spatially constrained clustering of image pixels, such that similar pixels are grouped into homogeneous and connected segments, called superpixels. Superpixels capture redundancy in an image and provide local image features that reduce the complexity of the image representation. For this reason, superpixels have become a common component in many recent frameworks involving image processing [Chu et al., 2015; Rehman et al., 2019; Subudhi et al., 2021].

Several superpixel algorithms have been developed based on different clustering principles, such as Quick Shift [Vedaldi and Soatto, 2008], Watersheds [Vincent and Soille, 1991], and Normalized cuts [Shi and Malik, 2000]. Two popular approaches are the k-means clustering-based *Simple Linear Iterative Clustering*

(SLIC) algorithm [Achanta et al., 2012] and the *Felzenszwalb's efficient graph based segmentation* [Felzenszwalb and Huttenlocher, 2004]. Both these methods approach the spatial constraining of the problem by representing each pixel by its intensity *and* spatial location. For a typical RGB image, this means that each pixel is represented by a vector (x, y, r, g, b), where (x, y) indicate its spatial location and (r, g, b) are the red, green, and blue channel intensities, respectively. The clustering is then performed in the five-dimensional feature space in order to group the pixels based on their intensity similarity and spatial proximity.

SLIC

The SLIC algorithm [Achanta et al., 2012] initializes k cluster centers spread in a grid across the image and is based on the k-means clustering algorithm, with two important modifications: i) The search space for each cluster center is limited to a squared search space proportional to the initial supervoxel size. ii) The distance measure between a sample $\boldsymbol{x} = (x_i, y_i, r_i, g_i, b_i)^T$ and a cluster center $\boldsymbol{\mu} = (x_j, y_j, r_j, g_j, b_j)^T$ is split into an intensity distance d_c:

$$d_c = \sqrt{(r_i - r_j)^2 + (g_i + g_j)^2 + (b_i - b_j)^2}, \tag{4.1}$$

and spatial distance:

$$d_s = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \tag{4.2}$$

which are and combined as a weighted sum to produce the final metric:

$$d_{SLIC} = \sqrt{d_c^2 + (\frac{d_s}{S})^2 m^2}, \tag{4.3}$$

where m balances the importance between spatial proximity and intensity proximity (with large values of m favouring compact superpixels), and S is set to the sampling interval of the initialization to normalize the spatial proximity.

Felzenszwalb

The Felzenszwalb algorithm [Felzenszwalb and Huttenlocher, 2004] is a graph-based clustering algorithm. The image is represented as a graph $G = (V, E)$, where the vertices $v \in V$ represent the individual pixels, and edges between connected vertices $(v_i, v_j) \in E$ are associated with a weight $w(v_i, v_j)$, representing the dissimilarity between v_i and v_j in the (x, y, r, g, b)-space. The clustering is performed in an agglomerative manner, starting with each vertex being its own segment C. Starting with the edge corresponding to the smallest

weight, a *predicate* D determines whether or not there is evidence for a boundary between the segments C_1 and C_2. If there is no evidence, the segments are merged. The predicate is designed by combining two measures: i) The internal difference of the components, computed as the largest weight within the minimum spanning tree (MST)[2] of the component:

$$\text{Int}(C) = \max_{e \in MST(C,E)} w(e). \tag{4.4}$$

ii) The difference between two components, defined as the smallest weight connecting the two components:

$$\text{Dif}(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w(v_i, v_j). \tag{4.5}$$

The predicate D compares two segments C_1 and C_2, by comparing their difference given by Equation 4.5 and their *minimum* internal difference, as:

$$D(C_1, C_2) = \begin{cases} \text{True}, & \text{if Dif}(C_1, C_2) > \text{MInt}(C_1, C_2) \\ \text{False}, & \text{else} \end{cases}, \tag{4.6}$$

where the minimum internal difference MInt is given by:

$$\text{MInt}(C_1, C_2) = \min\left(\text{Int}(C_1) + \tau(C_1), \text{Int}(C_2) + \tau(C_2)\right). \tag{4.7}$$

Here $\tau(C)$ is a threshold function that controls how strong the evidence must be for D to be True, and is commonly proportional to $|C|^{-1}$ to avoid many small segments. The merging continues until all pairs of segments satisfy the predicate.

The superpixels produced by these two algorithms can vary a lot in appearance, with the k-means based SLIC superpixels typically being more compact and equal-sized, and the Felzenszwalb superpixels being more irregularly shaped. Figure 4.2 shows an example image and its superpixel representation for SLIC superpixels and Felzenszwalb superpixels for three different superpixel resolutions. Note that for both algorithms, the extension to 3D supervoxels is trivial.

4.2 Few-shot Learning

Few-shot learning (FSL) is a recent and growing research field within deep learning that aims to learn models that can adapt to new concepts when provided with only a few labeled samples. Generally, there are two approaches

2. The MST of a component C is the subset of the edges $e \in E$ that connects all vertices in C together, with the smallest possible total edge weight.

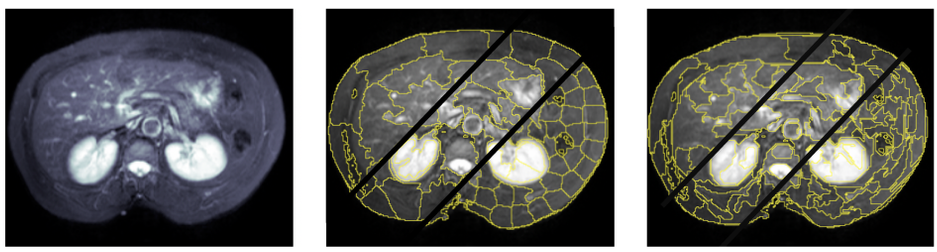


Figure 4.2: Examples of superpixel segmentations of an abdominal MRI slice from the CHAOS dataset [Kavur et al., 2021]. (a) Original image slice, (b)SLIC superpixel segmentations for three different resolutions (separated by black lines), and (c) Felzenszwalb superpixel segmentations for three different resolutions (separated by black lines).

to FSL: *Data-level* methods and *model-level* methods. The data-level methods are conceptually simple and aim to increase the labeled sample size to enhance generalization by, for instance, generating [Edwards and Storkey, 2016; Rezende et al., 2016], or hallucinating [Hariharan and Girshick, 2017; Gui et al., 2021] additional labeled examples. Model-level methods represent the larger group of techniques and mainly concerns *meta-learning* approaches, but also non-meta learning approaches, such as fine-tuning [Chen et al., 2019] and transductive fine-tuning [Dhillon et al., 2020; Ziko et al., 2020].

4.2.1 Few-shot meta-learning

Meta-learning, or *learning to learn*, is the most common approach to FSL and also the most relevant approach for this thesis. Few-shot meta-learning mimics the way humans learn new tasks, e.g. not slowly from scratch, but quickly by relying on experience from previous, related tasks [Vanschoren, 2018]. To achieve this property, the models are trained in episodes on a series of training tasks, where each task is a FSL problem. Specifically, in an N-way k-shot classification problem, the goal is to classify an unlabeled query image into one of N classes, based on k labeled support images per class. Based on the performance on the query image, a loss is computed and used to update the network. Once the training is done, the model is able to quickly adapt to the new testing tasks by only observing a few labeled instances.

Meta-learning approaches to FSL can roughly be divided into three categories: i) *Metric-based* methods that aim to learn a discriminative embedding space where samples from the same class are closely embedded and samples from different classes lie far apart. ii) *Optimization-based* methods that learn to fine-tune on a few samples by learning a good model-initialization that can adapt

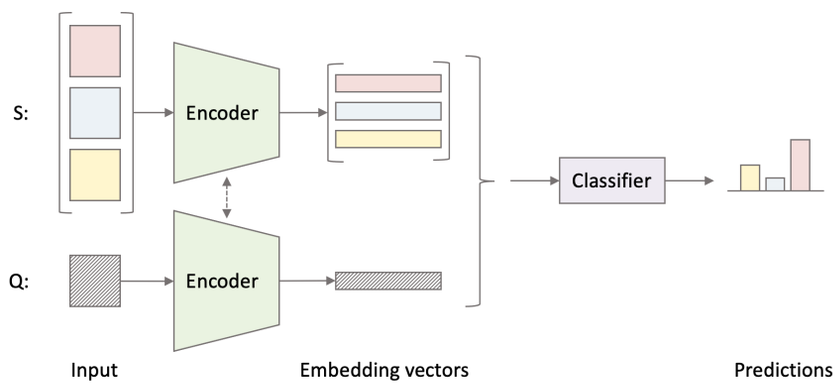


Figure 4.3: Illustration of a 3-way 1-shot classification problem in the general metric-based FSL framework. The support set (S), consisting of one labeled shot per class (red, blue, and yellow), is encoded to compute class-prototypes. The query image (Q) is encoded and, in this case, classified to the red class. The classifier can be a simple nearest-neighbour classifier or a parameterized neural network.

to new classes with a few gradient-based updates. iii) *Model-based* methods that develop architectures that map the query image and the support set to the query label in an end-to-end manner.

Having a simple and efficient design, the metric-learning-based approaches have received a lot of attention, especially in the extension from image classification to image segmentation, which is the focus in this thesis. For a comprehensive review of FSL, please refer to [Parnami and Lee, 2022].

Metric-learning-based FSL

The aim of metric-learning, or embedding-learning, approaches to FSL is to learn a mapping from the input to an embedding space where a defined metric (such as Euclidean distance, cosine distance, or a metric learned by a neural network) yields high distances between samples from different classes and low distances between samples from the same class. The general framework for metric-learning-based FSL is illustrated in Figure 4.3.

An important work in metric-based FSL is the *Prototypical Network* [Snell et al., 2017], which is a simple model based on the clustering assumption. The idea is that there exists an embedding space where samples cluster around their global class prototype, and that the classification problem can be reduced to a nearest-prototype classification in this embedding space. The framework consists of a network $f_{\boldsymbol{\theta}} : \mathbb{R}^d \rightarrow \mathbb{R}^m$, mapping the input $\boldsymbol{x} \in \mathbb{R}^d$ to the m-

dimensional feature space. In an N-way k-shot learning problem, the k labeled support samples from each class c, $\mathcal{S}_c = \{(\boldsymbol{x}_i, y_i), \ldots, (\boldsymbol{x}_k, y_k)\}$ are used to compute the class-prototype as:

$$\boldsymbol{p}_c = \frac{1}{k} \sum_{\boldsymbol{x}_i \in \mathcal{S}_c} f_\theta(\boldsymbol{x}_i). \tag{4.8}$$

Based on the N class-prototypes, the prediction of a query sample $\boldsymbol{x}^*$ is given by the softmax over the distances to the prototypes:

$$p(y = c|\boldsymbol{x}^*) = \frac{\exp(-d(f_\theta(\boldsymbol{x}^*), \boldsymbol{p}_c))}{\sum_{c'} \exp(-d(f_\theta(\boldsymbol{x}^*), \boldsymbol{p}_{c'}))}, \tag{4.9}$$

where $d(\cdot, \cdot)$ is a distance function. The model is trained in episodes by minimizing the cross-entropy loss via stochastic gradient descent.

This simple and efficient framework has inspired a whole line of work within few-shot classification, e.g. [Sung et al., 2018; Doersch et al., 2020; Kang et al., 2021; Afrasiyabi et al., 2022], *and* few-shot segmentation. Wang et al. [2019] adopted the Prototypical Network's simple framework to perform few-shot *segmentation*, the direct extension of few-shot classification to few-shot *pixel-wise* classification. A network $f_\theta : \mathbb{R}^{H \times W \times d} \to \mathbb{R}^{H' \times W' \times m}$ maps input images $\boldsymbol{X}$ to feature maps in an m-dimensional space, where *masked average pooling* is used to compute class-wise prototypes. In an N-way k-shot learning problem, the k labeled support samples from each class c, $\mathcal{S}_c = \{(\boldsymbol{X}_i, \boldsymbol{Y}_i), \ldots, (\boldsymbol{X}_k, \boldsymbol{Y}_k)\}$ are used to compute the foreground class-prototype as:

$$\boldsymbol{p}_c = \frac{1}{k} \sum_{(\boldsymbol{X}_i, \boldsymbol{Y}_i) \in \mathcal{S}_c} \frac{\sum_{x,y} f_\theta(\boldsymbol{X}_i)(x, y) \boldsymbol{Y}_i^c(x, y)}{\sum_{x,y} \boldsymbol{Y}_i^c(x, y)}, \tag{4.10}$$

where (x, y) indicate the spatial location and $\boldsymbol{Y}_i^c = \mathbb{1}(\boldsymbol{Y}_i == c)$ is the binary ground truth mask of class c. To compute the background prototype, the background feature vectors from all support samples are pooled:

$$\boldsymbol{p}_{bg} = \frac{1}{Nk} \sum_c \sum_{(\boldsymbol{X}_i, \boldsymbol{Y}_i) \in \mathcal{S}_c} \frac{\sum_{x,y} f_\theta(\boldsymbol{X}_i)(x, y) \boldsymbol{Y}_i^{bg}(x, y)}{\sum_{x,y} \boldsymbol{Y}_i^{bg}(x, y)}, \tag{4.11}$$

where $\boldsymbol{Y}_i^{bg} = \mathbb{1}(\boldsymbol{Y}_i == 0)$ is the binary ground truth mask of the background. Then the pixel-wise classification of the query image $\boldsymbol{X}^*$ is performed based on the closest prototype:

$$p(\boldsymbol{Y} = c|\boldsymbol{X}^*) = \frac{\exp(-\alpha d(f_\theta(\boldsymbol{X}^*), \boldsymbol{p}_c))}{\sum_{c'} \exp(-\alpha d(f_\theta(\boldsymbol{X}^*), \boldsymbol{p}_{c'}))}, \tag{4.12}$$

where $d(\cdot, \cdot)$ is the cosine distance and $\alpha = 20$ is a scaling factor. The model is trained in episodes, similarly to the Prototypical Network, but in addition

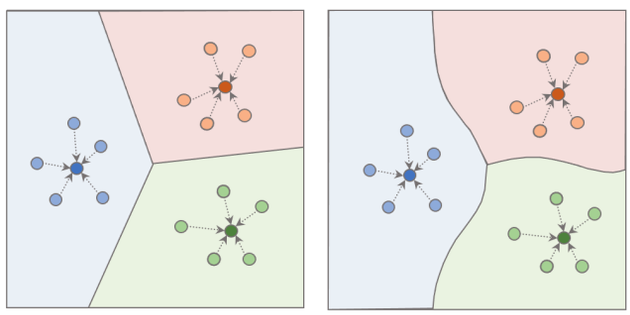


Figure 4.4: Illustration of different prototypical few-shot classifiers dividing the embedding space into two foreground classes (red and blue) and one background class (green). (a) Linear classifier (PANet). (b) Non-linear classifier (Class-Agnostic Segmentation Network (CANet)).

to the cross-entropy loss, a novel *prototype alignment regularization (PAR)* loss is used to update the weights. The PAR is performed by reversing the roles of the support and query. The predicted query segmentation is used as support to segment the original support images (now acting as query images), and gives name to the framework: Prototype Alignment Network (PANet). The PAR loss is then computed as the cross-entropy loss of this "reverse" segmentation, and the total loss is given by the sum of the regular cross-entropy loss and the PAR loss.

Concurrently to PANet, Zhang et al. [2019b] proposed the CANet. Similarly to PANet, CANet encodes the input $\boldsymbol{X}$ with a network $f_{\boldsymbol{\theta}} : \mathbb{R}^{H\times W\times d} \rightarrow \mathbb{R}^{H'\times W'\times m}$ to an m-dimensional feature space where masked average pooling is used to extract class prototypes. However, instead of classifying the query feature vectors by considering the cosine distances to the prototypes, CANet *learns* a non-linear distance function as a parametric network $g_{\boldsymbol{\phi}} : \mathbb{R}^{H'\times W'\times 2m} \rightarrow \mathbb{R}^{H'\times W'\times 2}$. The prototype $\boldsymbol{p}_c \in \mathbb{R}^m$ is concatenated with *all* spatial positions in the feature map $f_{\boldsymbol{\theta}}(\boldsymbol{X}^*) \in \mathbb{R}^{H'\times W'\times m}$, and the resulting tensor is decoded through $g_{\boldsymbol{\phi}}$[3]. The final segmentation result is obtained by up-sampling the output to original size (H, W), via bi-linear up-sampling. The use of a complex learnable distance function $g_{\boldsymbol{\phi}}$ introduces many additional learnable parameters, and to reduce the total number of parameters, CANet relies on pre-trained weights in the encoder $f_{\boldsymbol{\theta}}$ and does *not* update these during training. The parameters in g are updated by minimizing the cross-entropy loss between the query ground truth and the predicted mask. Figure 4.4 illustrates the difference between the

3. Note that the number of channels increase by m every time a prototype is concatenated with the feature map. Further, for N-way > 1, the order of the concatenation will effect the segmentation result. Models that depend on this type of dense comparison between features and prototypes therefore tend to focus on binary $N = 1$-way segmentation.

classifiers in PANet and CANet. With a parameterized classifier, CANet provides the opportunity to model more complex relations in the embedding space, but at the same time increases the risk of over-fitting.

Following PANet and CANet, different approaches have addressed various limitations of these frameworks. For instance, several improvements have been proposed to preserve more diverse features in the prototype extraction by computing multiple prototypes [Liu et al., 2020; Li et al., 2021; Zhang et al., 2021a; Yang et al., 2020], and to increase robustness in the feature comparison process by incorporating attention mechanisms [Zhang et al., 2019a; Wang et al., 2020]. Recently, there has been a larger focus on developing complex architectures, and state-of-the-art few-shot segmentation models typically rely on frozen pre-trained encoders [Zhang et al., 2021c; Min et al., 2021; Kang and Cho, 2022; Tian et al., 2020], making the adoption of these models to new domains challenging.

4.3 Self-supervised Learning

Self-supervision is a learning framework within the unsupervised learning paradigm, where a *pretext task* is defined such that the label information is implicitly available from the data. To be effective and encourage meaningful learning, a pretext task should require high-level image understanding to be solved.

The self-supervised learning technique was originally introduced in the text domain, but has proven to be a powerful tool in image processing as well. Common pretext tasks include relative patch location prediction [Doersch et al., 2015], image rotation prediction [Komodakis and Gidaris, 2018], and solving jigsaw puzzles [Noroozi and Favaro, 2016]. Recently, the focus of self-supervision has shifted from predicting these transformation properties towards learning *invariance* to such transformations [Misra and Maaten, 2020; Chen et al., 2020; He et al., 2020] (Illustrated in Figure 4.5). In particular, invariant representation learning by *contrastive learning* has gained popularity and most state-of-the-art self-supervision tasks rely on it [Chen et al., 2020; He et al., 2020; Chen and He, 2021]. The core idea in contrastive learning is to generate two *views* (a positive pair) of each image by applying diverse data augmentation techniques, and encourage the representations of these images to be similar, while at the same time being dissimilar to the representations of a set of other (negative) images. Contrastive learning is mostly explored for the classification task by learning consistent representations for image feature *vector* representations. However, some recent works also focus on dense contrastive learning designed for the segmentation task [Wang et al.,

2021b, 2022].

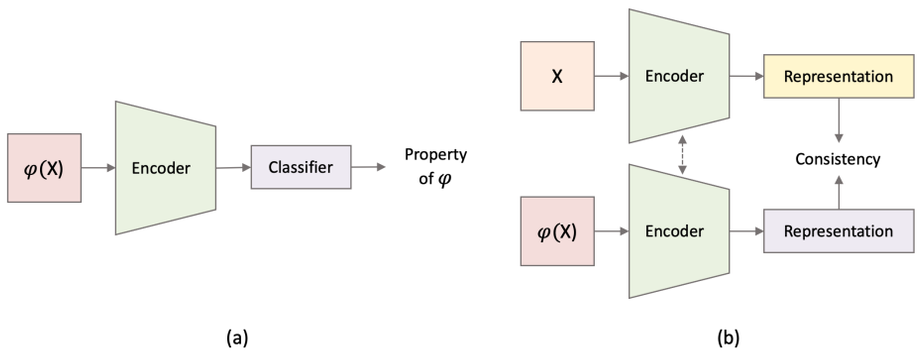


Figure 4.5: Two approaches to self-supervised representation learning. (a) Learning to predict the properties of a transformation. (b) Learning invariance to transformations.

Part II

Medical Image Segmentation with Limited Supervision

5

Medical Image Data

The field of medical imaging began with the discovery of the X-ray in 1895, for the first time allowing medical doctors to see inside a patient's body non-invasively. Today, medical imaging is an integral part of healthcare and plays a central role in, for instance, screening, diagnosis, treatment planning, and follow-up [Bercovich and Javitt, 2018]. Since the first X-ray image was taken some 120 years ago, the field has revolutionized medicine and now includes advanced imaging acquisition systems with various imaging modalities, capable of capturing specific types of anatomical information. The included works in this thesis concern segmentation of MRI and hybrid PET/MRI images. This chapter briefly introduces the principles of these modalities and discusses some of the challenges related to working with this type of data.

5.1 Principles of MRI and PET Imaging

MR Imaging MRI is one of the most used imaging techniques in medicine and provides high-resolution 3D visualization of internal structures in the body. MRI exploits strong magnetic fields and uses pulses of electromagnetic waves to excite water molecules in the body, then records the locations of the re-emitted waves with high precision. The relaxation of the molecules consists of two independent processes, longitudinal relaxation and transverse relaxation, generating T_1 and T_2 MRI images, respectively [Sun et al., 2008]. Different tissue

types have properties that result in different relaxation times, and the contrast in the MRI images are based on evaluating these local variations [Wallyn et al., 2019].

MRI is a non-invasive imaging technique and has become an important tool in diagnosis of many diseases, including cancer, stroke, brain disorders, and different types of heart conditions [Salzer, 2012].

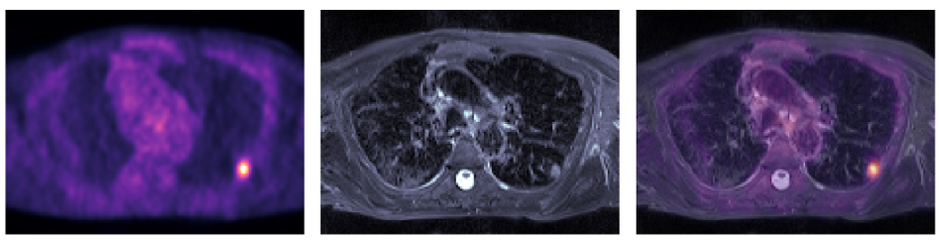
PET Imaging PET imaging is an important imaging technique that provides 3D functional information about the distribution of a tracer in the body [Iniewski, 2009]. A radioactive tracer with short half-life (in the order of minutes to hours) is injected into the blood stream, and as the tracer isotopes decay, the resulting gamma radiation is recorded. The voxel values in a PET image quantify the concentration, typically in units of standard uptake value (SUV), of the injected tracer. Depending on the properties of the tracer, PET imaging can visualize different processes in the body, such as metabolism and blood flow. In oncology, the accumulation of ^{18}F-fluorodeoxyglucose (FDG) provides the opportunity to measure the glucose consumption rate, which typically is high in malignant tumors [Crișan et al., 2022]. FDG-PET is therefore often used in diagnosis, staging, and monitoring of cancer patients [Iniewski, 2009].

Hybrid imaging Hybrid imaging, or combined imaging, refer to imaging systems that combine the acquisition of multiple complementary imaging modalities. A recent advancement in hybrid imaging is the PET/MRI scanner [Judenhofer et al., 2008], where the functional information from PET is anatomically localized using high spatial resolution MRI images. Compared to the more established hybrid modality PET/CT, PET/MRI has favourable properties in that MRI does not involve harmful radiation, is a more versatile imaging technology, and provides excellent soft-tissue contrast [Beyer et al., 2011].

Figure 5.1 provides an example illustrating the different individual strengths of PET and MRI, and how they complement each other to give co-registered functional and anatomical information when combined.

5.2 Data challenges

Apart from the challenging job of collecting and labeling medical images, working with these data comes with some additional challenges. Specifically, when working with the task of image segmentation, relevant challenges in-



(a) (b) (c)

Figure 5.1: Examples of (a) PET, (b) MRI, and (c) fused PET/MRI scans of a patient with lung cancer, collected in [Kuttner et al., 2020] using the ^{18}F-FDG tracer. These examples illustrate the individual strengths of the two modalities and how they combined provide co-registered information about the tumor volume (highlighted by PET) and the surrounding anatomical structures (provided by MRI).

clude:

Data size The size (in number of voxels) of a medical image volume depends on its resolution (voxel spacing) and the size of the imaged region (field of view). After pre-processing, a typical MRI image volume considered in this thesis is around $256 \times 256 \times 35$ voxels. Assuming that the voxels have floating point precision (32 bits/voxel) and that there are 30 such volumes in the dataset, this sums up to approximately $n = 68.8$ million voxels occupying over 275 Megabyte of memory. Depending on the chosen approach, processing this data can be challenging. For instance, among the clustering approaches considered in Paper I, spectral clustering does not scale well to this high number of samples as it requires the computation of the eigen-decomposition of a $n \times n$ Laplacian matrix. Furthermore, depending on the complexity of the architecture, processing such image volumes one-by-one in a deep-learning framework might also lead to GPU memory-issues as the encoding during training involves storing of intermediate activations.

Artifacts Different imaging techniques are subject to various types of artifacts during image acquisition, such as metal artifacts in MRI and tracer-related artifacts in PET [Simpson et al., 2017]. A general artifact, related to the spatial voxel resolution, is the *partial volume effect*. This occur whenever a single voxel covers multiple tissue types, resulting in blurry boundaries between different structures. MRI images are often acquired with an an-isotropic voxel resolution, leading to a more severe effect along the axis of lower resolution [Pham et al., 2000]. Figure 5.2 provides examples that illustrate vague, unclear boundaries between organs in MRI. Another artifact that increase the difficulty of the image segmentation problem is *motion blur* caused by patient movement during

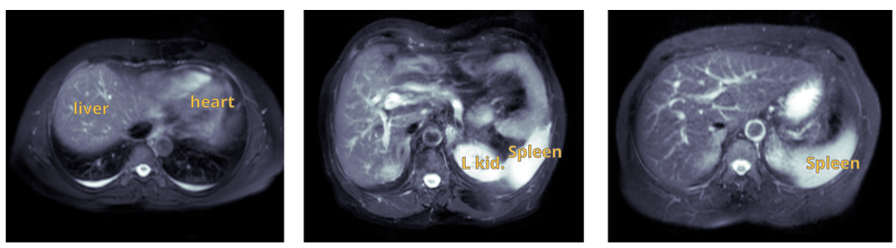


Figure 5.2: Examples illustrating unclear boundaries between organs in MRI slices from the CHAOS dataset [Kavur et al., 2020]. (a) Vague boundary between left liver lobe and heart. (b) Vague boundary between left kidney (L kid.) and spleen. (c) Vague boundary between spleen and surrounding tissue.

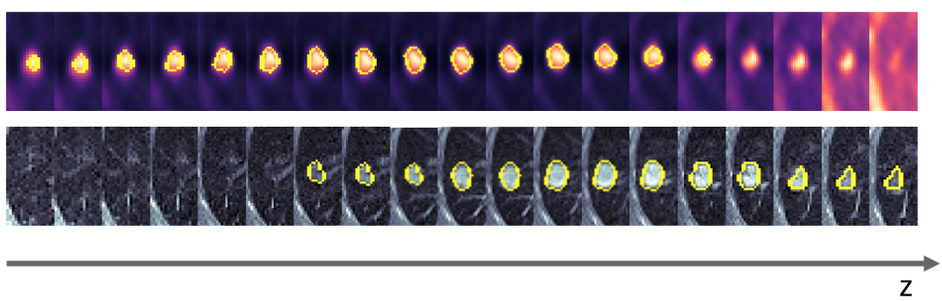


Figure 5.3: Example illustrating imperfect co-registration between simultaneously acquired PET (top) and MRI (bottom), collected in [Kuttner et al., 2020]. The figure displays axial-plane crops around the tumor volume (delineated in yellow) along the z-axis.

acquisition. In particular, imaging of motion-affected organs such as the lungs and the heart is challenging, and the longer the acquisition time is, the more severe the effect gets.

Co-registration Hybrid imaging systems, where multiple image modalities are acquired simultaneously, offer the best possible alignment of modalities. However, due to patient motion, such as respiratory motion, one can not assume perfect co-registration in hybrid imaging modalities. For instance, in PET/MRI, the typically longer PET acquisition time leads to more severe motion blur in PET compared to MRI, resulting in imperfect co-registration. This is particularly challenging in the imaging of small lung tumors, as observed in Paper I. Figure 5.3 illustrates an example where respiratory motion leads to imperfect co-registration of the lung tumor volume between simultaneously acquired PET and MRI.

6

Medical Image Segmentation

Image segmentation is a first step in many image analysis applications, as a means of simplifying the image representation before further analysis. Today's clinical practice is based on manual slice-by-slice image segmentation by trained physicians [Fu et al., 2021]. This work is tedious and time-consuming, and the resulting segmentation maps are prone to subjective interpretation [Nelms et al., 2012]. Moreover, the amount of medical image data collected at the hospitals is ever-growing, and a study examining the radiologists' workload showed that the number of images to be interpreted per radiologist per minute increased from 2.9 to 16.1 in a US hospital over a ten-year period [McDonald et al., 2015]. The investigation of automatic medical image segmentation is therefore of increasing importance and has developed into a broad field of research. Common approaches to medical image segmentation include [Pham et al., 2000]:

1. **Thresholding.** Thresholding techniques build on the simple principle that images can be segmented by grouping voxels based on their intensities. The process consists of determining one or more thresholds and can be done automatically [Feng et al., 2017] or through manual interaction, which is common practice in tumor segmentation from PET images (e.g. 40 % of the maximum SUV within a manually defined region of interest) [Mercieca et al., 2018].

2. **Region-based methods.** In region-based segmentation, a region is defined by a set of neighbouring voxels that abide some predefined criterion. In region-growing algorithms, the idea is to start with a manually selected seed voxel and let the region grow by considering the neighbouring voxels: If a neighbouring voxel meets the criterion it is merged into the region and the growing continues. Variants of region-growing algorithms are often used in tumor segmentation [Day et al., 2009; Dehmeshki et al., 2008; Wu et al., 2008].

3. **Deformable models.** Deformable models, or active contours/surfaces, fit closed parametric curves/surfaces to the perimeter of a structure. Starting by initializing the curve/surface close to the structure of interest, the curve iteratively evolves to minimize its energy, given by a function of a set of defined internal and external forces [Pham et al., 2000]. Deformable models typically also include a smoothness constrain, making them more robust to noise, and are used in various medical segmentation tasks [Morais et al., 2017; Rahmati et al., 2012; Rebouças Filho et al., 2017].

4. **Atlas-based methods.** Atlas-based methods rely on the availability of one (or more) atlas(es) and typically consider the segmentation problem as a registration problem, where the labeled atlas is co-registered to the image to be segmented. Atlas-based approaches have extensively been used for brain segmentation in MRI images [Lötjönen et al., 2010; Makropoulos et al., 2014; Iglesias et al., 2013].

5. **Clustering-based methods.** Clustering algorithms can be used to segment unlabeled medical images through voxel level grouping. To encourage spatially smooth clustering results in the presence of noise, efforts to include contextual information through Markov random fields [Chen et al., 2017b; Daniels and Gallagher, 2017] and superpixels [Kumar et al., 2019; Ye et al., 2010] have been investigated.

6. **Deep learning based methods.** Deep learning methods extract task-dependent representative features directly from the data and learn to segment images by either classifying [Ronneberger et al., 2015; Dong et al., 2021] or clustering [Ahn et al., 2021; Zhao et al., 2020] the image voxels.

The focus of his thesis is on methods to medical image segmentation which only require limited supervision and that do *not* rely on manual input during inference (like for instance manually selected seed points or regions of interest). In particular, Paper I concerns image segmentation by *clustering* and Papers II-III develop new *deep learning* methodology for few-shot medical image segmentation. This chapter introduces the segmentation tasks considered in Paper

I-III and provides a brief overview of relevant approaches to solve them.

6.1 Lung tumor segmentation

Lung cancer is the leading cause of cancer death and represented more than one in 10 cancers diagnosed in 2020 [Sung et al., 2021]. As part of the treatment planning and evaluation of therapy response of these patients, the accurate delineation of the tumor volumes is an extremely important, but inherently difficult task [Velazquez et al., 2013]. Firstly, the size, location, shape, and appearance of the tumors vary from patient to patient. Secondly, depending on the acquisition time during imaging, respiratory motion leads to motion blur in the images, causing tumors to appear elongated in the direction of movement. Further, in hybrid imaging where multiple imaging modalities are acquired simultaneously but with different acquisition times, this movement can damage the co-registration of the anatomical structures and the tumors, leading to a voxel-wise mismatch between images. In other words, lung tumor segmentation from PET/MRI is a difficult task with many challenges.

Given the relatively recent introduction of PET/MRI into clinical practice, the number of works on PET/MRI-based tumor segmentation is limited, with little focus on lung tumors in particular. In [Bagci et al., 2013] and [Xu et al., 2015], the authors propose tumor segmentation approaches that are evaluated on, amongst others, lung tumors. Bagci et al. [2013] propose a random walk-based co-segmentation of PET/MRI by computing a combinatorial hybrid Laplacian matrix as the product of the individual PET and MRI Laplacians. The authors further develop a seed-selection mechanism, thereby fully automating the process. Later, Xu et al. [2015] followed up this work by formulating a novel affinity function in a fuzzy connectedness segmentation algorithm, achieving similar results but with heightened efficiency.

Other works on tumor segmentation from PET/MRI, include pancreatic, liver, and prostate tumor segmentation [Sbei et al., 2017, 2020], and head-and-neck tumor segmentation [Leibfarth et al., 2015].

The work in Paper I focuses on the challenging task of unsupervised lung tumor segmentation from hybrid PET/MRI. As opposed to previous works, that segment PET/MRI image pairs separately, this paper performs the segmentation on a population-level, thereby providing the algorithms with all the available information. Furthermore, the work in Paper I has a special focus on lung tumors, analysing the algorithm's sensitivity to, amongst others, tumor size and tumor-overlap in the modalities.

6.2 Organ segmentation

Organ segmentation from MRI is a common task in the medical domain and consists in segmenting organs from their surrounding tissues (e.g. segmenting abdominal organs such as kidneys, liver and spleen), or segmenting organs into their constituent parts (e.g. segmenting the chambers in the heart). The resulting segmentation maps hold valuable information that are used by the medical doctor to e.g. help identify organs at risk prior to radiotherapy [Chen et al., 2021], or quantize tissue volumes for diagnosis [Schick, 2022]. A recent and promising field of research focuses on the development of data-efficient automatic organ segmentation through FSL and the methods discussed in Section 4.2.

Building on the PANet framework [Wang et al., 2019], different modifications and extensions have been proposed to meet the specific challenges and opportunities related to working with medical data. Yu et al. [2021] exploit the structural consistency between abdominal MRIs by enforcing strict spatial priors on the segmentation problem. Instead of segmenting the query image as one whole, it is divided into a grid and each element is segmented separately. Class-wise prototypes extracted from the corresponding grid element in the support image are used to guide the segmentation. Through this grid-structure approach, the authors achieve more *location sensitive* prototypes, which is beneficial especially for the large and inhomogeneous background class. However, a high grid-resolution, providing more high-quality background prototypes, comes at the cost of requiring highly standardized datasets with good alignment between support and query images, especially when the target structures are small[1]. To assure sufficient overlap between support and query, Yu et al. [2021] employ a grid size of $1/8$ of the image size, meaning that modelling the background with one prototype still is problematic. In a similar spirit, Tang et al. [2021] also build on PANet. By coupling a prototypical few-shot segmentation network (PANet) with a novel recurrent mask refinement module, the module iteratively refines the segmentation map through encoding relations between foreground and background.

Few-shot learning models, in general, are only few-shot in the sense that a trained few-shot model only requires a few labeled instances to segment a new class. Both Yu et al. [2021] and Tang et al. [2021] propose models that are trained in a *supervised* manner, meaning that they require a set of labeled training tasks that are different from the test tasks. As the availability of labeled datasets typically is limited in the medical domain, Ouyang et al. [2022] propose a self-supervised training task that allows for *unsupervised* training of

1. Achieving sufficient alignment between images becomes even more challenging in a potential 3D extension of this approach.

their model. The self-supervised training tasks are generated from unlabeled image slices by leveraging their superpixel segmentation. An image/label pair is generated from one unlabeled image slice and its corresponding superpixel segmentation by sampling one random superpixel as foreground class and letting the union of the remaining superpixels represent the background class. To simulate intensity/geometric variations between support and query, and encourage model invariance to these, random transformations are applied to a copy of the image and the label. The original image/label pair becomes the support image/label whereas the transformed copy represents the query image/label.

To address the challenge of modelling heterogeneous classes with single prototypes, Ouyang et al. [2022] propose the *adaptive local prototype pooling network (ALPNet)*. Instead of simply computing global foreground/background prototypes, the authors additionally compute a set of prototypes on a regular grid to preserve local information. Specifically, during inference, they employ a grid-size of 2×2, resulting in 256 local prototypes per image slice, most of which represent the background. Even though this approach captures more local information, it comes at the cost of computational complexity, as similarity maps must be computed for each prototype. This further complicates a potential 3D extension of the methodology.

All the methods discussed above approach the problem of medical image volume segmentation as a series of slice-wise 2D segmentations. This consequently demands an evaluation protocol describing a scheme for matching support and query slices during inference: *Which support slices should be used to extract prototypes and which query slices should these prototypes segment?* The standard evaluation protocol divides the target structures in both support and query images into *three* succeeding sub-chunks, then let the *middle* slice in each support sub-chunk guide the segmentation of all slices in the corresponding query sub-chunk. However, this protocol requires weak label information about the test data in order to locate the target structure. This means that the user, prior to segmentation, must search through the image volume slice-by-slice to mark slices containing the target structure.

The work in Paper II and III focuses on the task of MRI-based organ segmentation, specifically abdominal organ segmentation and cardiac segmentation, in the prototypical few-shot setting. Paper II, builds further on the self-supervision task by Ouyang et al. [2022] and develops new methodology to address the challenge of modeling the complex background class. The paper further introduces a more realistic evaluation protocol that does *not* require weak label information. Paper III follows up the work in Paper II by proposing new ideas to better exploit the available information during inference phase to achieve more accurate and more trustworthy predictions.

Part III

Summary of Research

7

Paper I

Unsupervised supervoxel-based lung tumor segmentation across patient scans in hybrid PET/MRI.

The main objective of this paper is to investigate the potential of using PET/MRI, a new hybrid imaging technology, for the difficult task of unsupervised lung tumor segmentation. We propose a method that does not require manual user input and that performs clustering *across* patient scans in a population-level manner to better exploit the patterns in all the available data. In particular, we explore a two-step approach consisting of a supervoxel-level feature extraction on the patient-level, followed by a population-level clustering into tumor and non-tumor supervoxels.

Firstly, as mentioned in Section 5.2, respiratory motion during the image acquisition process leads to voxel-wise mismatch between the PET and MRI scans. To alleviate the effect of the imperfect co-registration, all image pairs are co-registered prior to segmentation, using an automatic unsupervised framework.

The complete segmentation framework is illustrated in Figure 7.1 and starts with a supervoxel-based over-segmentation of all image volumes, followed by the extraction of two regional hand-crafted features from each supervoxel. As a final pre-processing step, the feature vectors are transformed with a Box-Cox

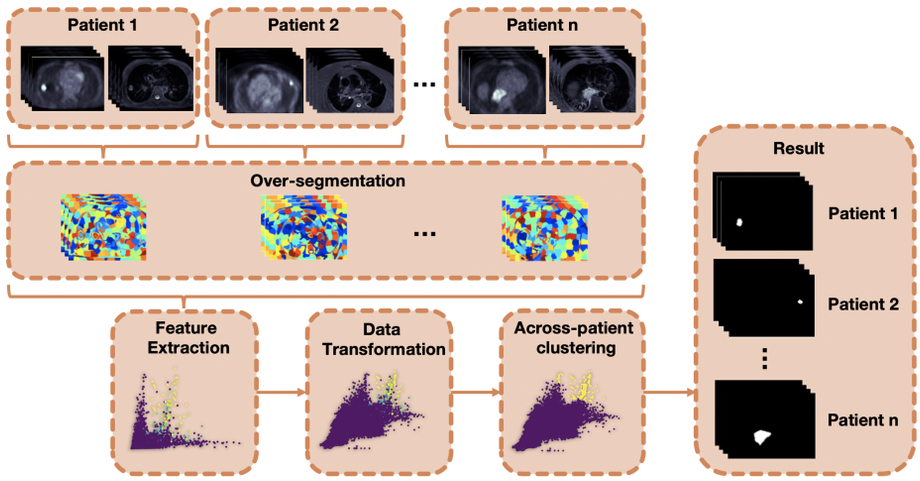


Figure 7.1: Illustration of the proposed unsupervised supervoxel-based lung tumor segmentation framework taken from Paper I. The two-stage approach consists in a supervoxel-clustering on patient-level, followed by a population-level clustering of supervoxel-extracted features.

transformation [Box and Cox, 1964] before being clustering into two clusters, representing tumor and non-tumor supervoxels.

As discussed in Section 5.2, not all machine learning algorithms scale well with respect to the number of data samples. To address this challenge and to reduce the influence of voxel noise, we work on the supervoxel-level instead of the voxel-level. The use of supervoxels reduces the computational cost in the clustering step of our proposed approach, by reducing the number of samples from more than 28.7 million voxels to less than 27k supervoxels, thereby making the proposed across-patient clustering approach feasible in practice.

Within the proposed segmentation framework, five variations of classical clustering algorithms are tested to evaluate their sensitivity to tumor size and voxel noise, in addition to analysing the type of segmentation mistakes they are prone to making and quantifying their associated benefit of the population-level clustering.

The results illustrate the benefit of clustering across patients, and an analysis of the errors made by the different clustering algorithms indicates that the segmentation of small-sized tumors in the presence of imperfect co-registration is particularly challenging. Furthermore, our study illustrates that spectral clustering achieves promising results and tends to be more robust to moderate levels of voxel-noise.

Contributions by the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

8

Paper II

Anomaly Detection-Inspired Few-Shot Medical Image Segmentation Through Self-Supervision With Supervoxels

In this paper, we propose an anomaly detection-inspired few-shot segmentation network that is trained self-supervised on supervoxel-derived pseudo-labels, and that only requires a few labeled samples during inference. By treating the problem as an anomaly detection problem, we bypass the challenge of explicitly modelling the large and inhomogeneous background class with prototypes. Figure 8.1 illustrates the difference between our proposed classifier and previous FSL approaches, as discussed in Section 4.2. Through our anomaly-detection inspired approach, we get the benefit of a non-linear classifier at the cost of only one extra learnable parameter (the threshold parameter). This also allows for a more relaxed background embedding, where feature vectors corresponding to different anatomical structures in the background class are *not* forced to belong to, and thereby cluster around, the same prototype.

By only explicitly modeling the foreground classes with prototypes, our model is less sensitive to a varying background class. This allows us to easily transition from the standard approach consisting of 2D slice-by-slice segmentation of the image volume, requiring weak label information to locate the target structures, to a more realistic approach where segmentation is performed directly in a one-step volume-wise manner.

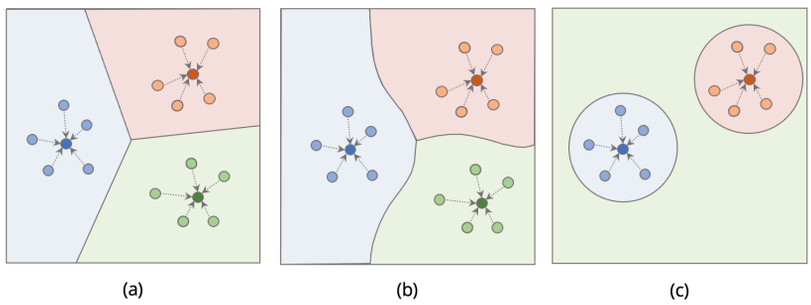


Figure 8.1: Different few-shot classifiers dividing the embedding space into two foreground classes (red and blue) and one background class (green). (a) Linear classifier (PANet). (b) Non-linear classifier (CANet) (c) Anomaly-detection inspired classifier (ADNet). Ouyang et al. [2022] address the problem with a linear classifier (like PANet), but with multiple prototypes per class.

Our proposed self-supervision task builds on the task proposed in [Ouyang et al., 2022], by extending it from super*pixels* to super*voxels*. This modification enables the utilization of the 3D nature of the medical images and allows for a more flexible sampling of support/query images during training-episode construction.

The self-supervision task shares similarities with contrastive learning, as discussed in Section 4.3, and can be seen as a dense contrastive learning task: The support and query images (to views) are generated from the same original image volume, by perturbing it through applying varying intensity and geometric transformations. The model is then trained to segment the query image based on the (pseudo) labeled support image, thereby aligning the feature representations of the foreground class and encouraging invariance to irrelevant variations in geometry/intensity.

The results demonstrate that the proposed anomaly detection approach to the problem yields a model that is robust to background outside the support slice, resulting in less over-segmentation. Experiments further indicate the benefit of extending the self-supervision task from superpixels to supervoxels, and illustrate the potential of the proposed methodology to perform one-step volume-wise segmentation.

An extended abstract (Paper 4) of preliminary results leading to this paper was presented at the Norwegian Society for Image Processing and Machine Learning (NOBIM) conference, Oslo, Norway in 2021. Further, a continuation of the work in this paper (Paper 8) was presented at the Colour and Visual Computing Symposium (CVCS), Gjøvik, Norway in 2022.

Contributions by the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

9

Paper III

ADNet++: A few-shot learning framework for multi-class medical image volume segmentation with uncertainty-guided feature refinement

This paper follows up the work in Paper II by improving the inference phase. Specifically, this paper proposes new methodology that better utilizes the available information during inference in order to produce more *accurate* and more *trustworthy* predictions.

The medical few-shot segmentation networks discussed in Section 6.2, including the anomaly detection-inspired network proposed in Paper II, follow the *encoder + classifier + up-sampling* approach to segmentation, discussed in Section 3.2.4. This means that the models perform the segmentation on spatially compressed feature representations, without any mechanism to alleviate the loss of spatial details during the image encoding process. To address this shortcoming, we propose a novel feature refinement module to help guide the precise location of edges in the segmentation map. To this end, we leverage the automatically generated supervoxel segmentation of the input image to refine the corresponding feature representation towards respecting the supervoxel edges. This is demonstrated to lead to more accurate segmentation maps. An illustration of the concept of the proposed supervoxel-informed feature refinement is provided in Figure 9.1.

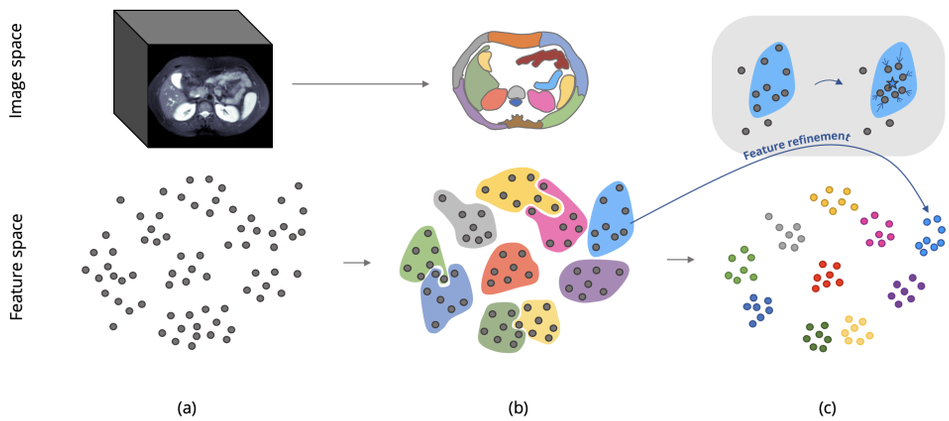


Figure 9.1: Conceptual illustration of the proposed supervoxel-informed feature refinement framework taken from Paper III. (a) The input image is embedded into a set of feature vectors (grey dots). (b) The supervoxels are used to identify feature vectors that "belong" together in the input-space. (c) To encourage a segmentation result that respects edges defined in the input space, the feature vectors are refined by moving them towards their corresponding supervoxel centers.

Further, to provide uncertainty maps and increase the trustworthiness of the model, we propose an architecture-agnostic mechanism, inspired by the Monte-Carlo dropout discussed in Section 3.3.2, that does not involve specific architectural requirements, such as dropout layers. We illustrate the fidelity of the resulting uncertainty maps and the potential of using these to guide the proposed feature refinement module.

Finally, current medical few-shot segmentation models are limited to binary segmentation, which can lead to ambiguous voxel predictions in multi-class segmentation problems. To avoid this problem, we develop a mechanism that performs multi-class segmentation in one step.

The proposed advancement of the inference phase leads to a model with significantly improved segmentation performance, compared to the method presented in Paper II. Furthermore, the model provides uncertainty maps with important information indicating which regions can and can not be trusted.

Contributions by the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.

- I wrote the original draft of the manuscript.

10

Concluding Remarks

The aim of this thesis was to develop new ML algorithms for medical image segmentation with limited supervision. In particular, the three main research objectives of the thesis were as follows:

1. Leverage data-specific opportunities in medical images through automatically generated supervoxels to train segmentation models with limited supervision.

2. Reconsider the current approach to few-shot medical image segmentation to obtain models that are robust to a large and inhomogeneous background class.

3. Design ad-hoc approaches to quantify the uncertainty in segmentation predictions and use this information to improve performance.

Through the work in this thesis, supervoxels are demonstrated to be versatile tools for leveraging structural information in medical data when training segmentation models with limited supervision. An approach to unsupervised lung tumor segmentation exploring the potential of clustering voxels *across* patient scans was developed in order to utilize the patterns in all the available data. To facilitate this population-level clustering, supervoxels were leveraged to reduce the computational complexity, thereby addressing research objective 1.

In the intersection between research objective 1 and 2, a self-supervised few-shot segmentation network was developed. Supervoxels were leveraged in a novel self-supervision task to train the network without requiring manually labeled images. Further, to obtain a segmentation model that is robust to the large and typically inhomogeneous background class, an anomaly detection-inspired classifier with a learnable threshold was developed.

In the intersection between research objective 1 and 3, a novel uncertainty-guided, supervoxel-informed feature refinement module was proposed for few-shot segmentation models. An architecture-agnostic mechanism to estimate uncertainty in few-shot segmentation networks was developed and demonstrated to produce uncertainty maps that quantify how much a prediction can be trusted. These uncertainty maps where further used to guide the proposed feature refinement model, in order to trust the initial prediction less in regions where the model is uncertain.

10.1 Limitations and Outlook

A general limitation of the methodology in this thesis is that the segmentation performance is directly connected to the quality of the generated supervoxels. While in this thesis the supervoxel hyper-parameters were found empirically, and showed to be relatively robust, alternative automated approaches or good heuristics would be beneficial. A potential solution could also be to explore multi-scale supervoxels [Tong et al., 2014] in the proposed frameworks to obtain more robust supervoxels.

Specifically for Paper I, the performance of the clustering relies on the extracted features being able to discriminate between the two classes. However, the features explored in the paper are in some cases insufficient and lead to overlap between the feature vectors corresponding to tumor and non-tumor regions. Alternative types of features that exploit supervoxel shape, texture, and histogram features could potentially increase discrimination. Similarly, deep learning could be leveraged to extract high-level features. Other limitations related to the data itself include the relatively small sample size and the imperfect co-registration. Even after the applied co-registration between PET and MRI, miss-matches are still present in the images, and alternative methods to co-registrate the images could potentially improve the results.

The classifier in Paper II and III is limited to modelling the target class with one single prototype. While this was demonstrated to achieve good performance in the tasks of abdominal organ segmentation and cardiac segmentation, it might be insufficient in tasks with less homogeneous foreground classes. To model

more complex target classes, a modification of the classifier with multiple foreground class prototypes obtained via e.g. clustering could be explored. Another approach to obtain a more robust classifier could be through a transductive inference fine-tuning [Boudiaf et al., 2021], where the learned threshold is updated based on the labeled support and the unlabeled query image in the specific inference episode.

In the proposed uncertainty-guided feature refinement module in Paper III, the uncertainty that is accounted for is the uncertainty in the model, while uncertainty in the supervoxel generation is ignored. An interesting future direction would be to jointly exploit both types of uncertainty to further improve the feature refinement.

Future directions In the past few years, ML-enabled devices have become increasingly prevalent in the medical field. However, the learning algorithms behind these devices are still limited to large fully annotated datasets. The work in this thesis has developed new methodology for medical image segmentation that is more label efficient, paving the way towards a more widespread applicability of ML-based solutions. However, multiple challenges still require attention in order to facilitate the use of these techniques in the clinic.

While the proposed methods provide promising predictive results, there is still a considerable gap to fully supervised approaches in terms of performance, and future work should aim to further reduce this gap. Towards this goal, new developments within self-supervised learning for segmentation [Cho et al., 2021; Dong et al., 2021; Wang et al., 2021b, 2022] are expected to be an important contributor.

Furthermore, the main focus of this thesis has been on the development of new ML methodology for medical image segmentation. As the development of the methodology progresses, the natural next step would be to target specific clinical applications in an interdisciplinary approach together with clinicians. In this extension, to build trust with the users, it is important to also focus on the explainability of the model predictions. Ideally, any prediction should be accompanied by an explanation that is meaningful to the user, providing additional information to increase trust in and safety of the system. Towards this goal, recent advances in the field of explainability could be exploited [Wickstrøm et al., 2020; Gautam et al., 2022a,b; Holzinger et al., 2022].

Part IV

Included Papers

Paper I

Unsupervised supervoxel-based lung tumor segmentation across patient scans in hybrid PET/MRI

Stine Hansen, Samuel Kuttner, Michael Kampffmeyer, Tom-Vegard Markussen, Rune Sundset, Silje Kjærnes Øen, Live Eikenes, and Robert Jenssen

Expert Systems with Applications, 2021

Expert Systems With Applications 167 (2021) 114244

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Unsupervised supervoxel-based lung tumor segmentation across patient scans in hybrid PET/MRI

Stine Hansen [a,*], Samuel Kuttner [c,d], Michael Kampffmeyer [a], Tom-Vegard Markussen [e], Rune Sundset [c,d], Silje Kjærnes Øen [b], Live Eikenes [b], Robert Jenssen [a]

[a] Department of Physics and Technology, UiT The Arctic University of Norway, NO-9037 Tromsø, Norway
[b] Department of Circulation and Medical Imaging, Faculty of Medicine and Health Sciences, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway
[c] PET Imaging Center, University Hospital of North Norway, NO-9038 Tromsø, Norway
[d] Department of Clinical Medicine, UiT The Arctic University of Norway, NO-9037 Tromsø, Norway
[e] University Hospital of North Norway, NO-9019 Tromsø, Norway

ARTICLE INFO

Keywords:
Clustering
Unsupervised learning
Medical image segmentation
Tumor segmentation
Hybrid PET/MRI

ABSTRACT

Tumor segmentation is a crucial but difficult task in treatment planning and follow-up of cancerous patients. The challenge of automating the tumor segmentation has recently received a lot of attention, but the potential of utilizing hybrid positron emission tomography (PET)/magnetic resonance imaging (MRI), a novel and promising imaging modality in oncology, is still under-explored. Recent approaches have either relied on manual user input and/or performed the segmentation patient-by-patient, whereas a fully unsupervised segmentation framework that exploits the available information from all patients is still lacking.

We present an unsupervised across-patients supervoxel-based clustering framework for lung tumor segmentation in hybrid PET/MRI. The method consists of two steps: First, each patient is represented by a set of PET/MRI supervoxel-features. Then the data points from all patients are transformed and clustered on a population level into tumor and non-tumor supervoxels. The proposed framework is tested on the scans of 18 non-small cell lung cancer patients with a total of 19 tumors and evaluated with respect to manual delineations provided by clinicians. Experiments study the performance of several commonly used clustering algorithms within the framework and provide analysis of (i) the effect of tumor size, (ii) the segmentation errors, (iii) the benefit of across-patient clustering, and (iv) the noise robustness.

The proposed framework detected 15 out of 19 tumors in an unsupervised manner. Moreover, performance increased considerably by segmenting across patients, with the mean dice score increasing from $\mathbf{0.169 \pm 0.295}$ (patient-by-patient) to $\mathbf{0.470 \pm 0.308}$ (across-patients). Results demonstrate that both spectral clustering and Manhattan hierarchical clustering have the potential to segment tumors in PET/MRI with a low number of missed tumors and a low number of false-positives, but that spectral clustering seems to be more robust to noise.

1. Introduction

Medical imaging is today an integrated part of diagnostics and treatment planning of cancer patients. In particular, hybrid positron emission tomography (PET)/computed tomography (CT) has become an established tool in tumor detection, characterization, staging, and monitoring (Flechsig, Mehndiratta, Haberkorn, Kratochwil, & Giesel, 2015; Ehman et al., 2017). A more recent advancement in hybrid radiologic imaging is the PET/magnetic resonance imaging (MRI) scanner, in which the anatomical information is obtained from MRI instead of CT. As opposed to CT, MRI does not involve harmful ionizing radiation and offers superior soft-tissue contrast with high spatial resolution, making PET/MRI a promising hybrid modality in, for instance, oncology. Nevertheless, the potential of hybrid PET/MRI is still being investigated and remains an open question (Ehman et al., 2017). This includes its potential in the important task of lung tumor segmentation,

* Corresponding author.
E-mail addresses: s.hansen@uit.no (S. Hansen), samuel.kuttner@uit.no (S. Kuttner), michael.c.kampffmeyer@uit.no (M. Kampffmeyer), tom-vegard.markussen@unn.no (T.-V. Markussen), rune.sundset@unn.no (R. Sundset), silje.k.oen@ntnu.no (S.K. Øen), live.eikenes@ntnu.no (L. Eikenes), robert.jenssen@uit.no (R. Jenssen).

https://doi.org/10.1016/j.eswa.2020.114244
Received 10 September 2020; Received in revised form 2 November 2020; Accepted 4 November 2020
Available online 29 November 2020

which is the focus of this paper.

Lung cancer is the most frequently diagnosed cancer type in the world, with a predicted number of 2.1 million new incidences in 2018 (Bray et al., 2018). An important, but inherently difficult, part of the treatment planning and follow-up of these cancerous patients is the process of isolating the tumor volume in medical images (Sauwen et al., 2016). Today, this tumor segmentation is commonly performed manually in a slice-by-slice manner. However, this work is tedious and susceptible to subjective interpretation (Caldwell et al., 2001; Hurkmans et al., 2001). A great amount of effort has therefore been put into the investigation of automatic tumor segmentation (Foster, Bagci, Mansoor, Xu, & Mollura, 2014; Gordillo, Montseny, & Sobrevilla, 2013).

The majority of existing methods for automatic medical image segmentation are based on *supervised* models that require fully annotated data sets to learn a classification of voxels into tumor and non-tumor voxels (De Bruijne, 2016). Such data sets are time-consuming to generate because segmentations have to be manually delineated for a large set of images. *Unsupervised* segmentation methods, on the other hand, have the benefit of not requiring annotations and is typically relying on voxel-wise clustering based on similarity within the data.

Only a few studies have considered tumor segmentation in hybrid PET/MRI (Bagci et al., 2013; Xu, Bagci, Udupa, & Mollura, 2015; Leibfarth et al., 2015; Sbei, ElBedoui, Barhoumi, Maksud, & Maktouf, 2017; Sbei, ElBedoui, Barhoumi, & Maktouf, 2020) (see related work section for details). In this paper, we aim to contribute to the recent line of work in order to further investigate the potential of PET/MRI for *unsupervised* lung tumor segmentation.

Unlike previous approaches to unsupervised tumor segmentation in hybrid PET/MRI, which perform segmentation in a patient-by-patient manner (Bagci et al., 2013; Sbei et al., 2020), we take advantage of the information in *all* available patient scans. In patient-by-patient segmentation approaches where the segmentation is based on single image pairs (PET and MRI from one patient), the number of voxels representing the tumor might be insufficient for the clustering algorithms to recognize them as a separate cluster. By instead clustering *across* patients in a population-level manner, we open up to taking advantage of the information in all patients when finding patterns to base the segmentation on. Voxel-wise clustering across all patients is, however, not computationally feasible as the total number of voxels becomes too high. To overcome this barrier, we take inspiration from a recent innovative approach to the problem of clustering tumor subvolumes (Wu et al., 2016; Even et al., 2017), by employing supervoxels rather than working directly on voxel level.

In our work, we thus examine a two-stage clustering approach for automatic lung tumor segmentation, where we first do a patient-level over-segmentation into homogeneous supervoxels, before we group the supervoxels *across all patients* and do a population-level clustering into "tumor" and "non-tumor" supervoxels. Since the problem at hand is complex and requires a systematic analysis of the proposed two-stage approach, we provide a comparison and analysis of several different clustering procedures to achieve this task. We further evaluate the advantage of utilizing across-patients information, the method's robustness to noise, the effect of tumor size and the types of segmentation errors.

The key contributions of this paper are:

1. A novel unsupervised lung tumor segmentation framework that can utilize information across patients in PET/MRI images.
2. An analysis of several commonly used clustering approaches within the proposed framework.
3. An analysis of the segmentation mistakes that the different clustering algorithms make and how tumor size affects the performance.
4. An analysis of the benefit of across-patients clustering compared to patient-by-patient clustering.
5. An analysis of the proposed method's sensitivity to image noise.

In the following, Section 2 provides a brief overview of the related work. Section 3 introduces the data set used as part of this study and Section 4 presents the proposed framework for lung tumor segmentation. In Section 5, the experimental results and an analysis of the segmentation mistakes, the effect of tumor size, the benefit of clustering across-patients, and a noise analysis are provided. Finally, Section 6 and Section 7 discuss outlook, limitations, and provide conclusions.

2. Related work

Today there exists a large range of methods for tumor segmentation in established modalities such as PET, CT, MRI and hybrid PET/CT (Foster et al., 2014; Moghbel, Mashohor, Mahmud, & Saripan, 2018; Wadhwa, Bhardwaj, & Verma, 2019; Ju et al., 2015), while the use of hybrid PET/MRI is less explored. In order to provide the necessary context to place this paper's contributions in the field, this section will highlight previous work within hybrid PET/MRI tumor segmentation. As this is a relatively new modality, previous studies are limited to only a handful of articles. To the authors' knowledge, the first study on tumor segmentation in hybrid PET/MRI was the study by Bagci et al. (2013), in which a random walk based co-segmentation approach with automatic foreground/background seed selection was developed. By unifying the graph representation of each modality in a single product lattice, they reformulated the random walk method to jointly delineate objects in different image modalities. A few years later, in the study by Xu et al., 2015, a tumor segmentation approach based on fuzzy connectedness with a visibility weighting scheme was proposed as a faster alternative achieving similar performance to Bagci et al. (2013). However, as opposed to Bagci et al. (2013), which performed segmentation unsupervised, the approach by Xu et al. (2015) required user-specified weights for each modality. Sbei et al. (2017) further developed the fuzzy connectedness approach and combined it with the graph cut method to address problems with leakage through weak boundaries. In recent work, Sbei et al. (2020) made additional modifications to the method by improving the automatic seed generation step and automatically generating intermediate images with reduced heterogeneity, which the segmentation is based on.

Common to all these approaches is that the segmentation is performed patient-by-patient. That is, only the information in the PET/MRI from one patient is considered at a time. A quite different approach was developed by Leibfarth et al. (2015), where tumor probability maps were derived for both PET and MRI images using heuristic probability mapping functions relating probability values and intensities on voxel level. Then the tumor delineation was derived using the threshold level set segmentation algorithm on the combined probability map defined by the weighted sum of the single maps. In this approach, the parameters were optimized by considering multiple patients in a supervised leave-one-out manner.

In our proposed framework, we depart from previous work and perform a PET/MRI tumor segmentation that is both *unsupervised* and exploits the information in *all available patient scans* by performing an across-patients clustering.

3. Dataset

For the current work, we used 18 PET/MRI acquisitions from a previous lung cancer study (Kuttner et al., 2020). The study was approved by the Norwegian Regional Committees for Medical and Health Research Ethics (REC reference 2017/915), and all patients signed written informed consent. The benefit of using these exams is that all scans contain one or multiple tumors diagnosed as either adenocarcinoma or squamous cell carcinoma, which are the two most common types of non-small cell lung cancer (Raponi et al., 2006).

Prior to PET/MRI, each patient was injected with 4 MBq/kg 18F-fluorodeoxyglucose (FDG). Approximately two hours post-injection, a 10-min, one-bed position PET acquisition of the mediastinum was

performed in a Siemens Biograph mMR (software version VB20P) (Siemens Healthineers, Erlangen, Germany) using a free-breathing and arms-down scan protocol. Simultaneous with PET, a T2-weighted TIRM MRI sequence was acquired. Furthermore, a standard DIXON-based MR sequence was used for attenuation correction of the PET images.

PET images were reconstructed using the ordered-subset expectation–maximization (OSEM) algorithm with three iterations, 21 subsets, and 4 mm Gaussian smoothing. For each PET image, the measured tissue radioactivity concentration [kBq/mL] was normalized against patient body weight and injected dose to obtain the standardized uptake value (SUV) [g/mL]. The gross tumor volume was delineated in the T2 images for all patients based on morphology. PET images were used as an aid to differentiate pathology from anatomy or atelectasis, or from large hilar vessels. Delineations were performed by a thorax radiologist ($> 10y$ experience) using Varian Eclipse Treatment Planning System version 10.0.42. In the PET images, FDG-avid lesions were segmented using a 41% SUVmax threshold. The union of the PET and MRI masks is considered the ground truth mask.

As a pre-processing step, the images were re-sampled to the same isotropic voxel resolution of $2 \times 2 \times 2$ mm^3 using cubic interpolation, resulting in an image size of $114 \times 152 \times 93$ voxels. More information about the data is summarized in Table 1. Fig. 1 shows two PET/MRI pair examples with corresponding ground truth masks.

4. Framework for lung tumor segmentation

Our proposed across-patients supervoxel-based clustering framework segments lung tumors in hybrid PET/MRI. Supervoxels are computed for each patient and features extracted from these are grouped. In order to improve the segmentation performance, the features are transformed using a Box-Cox transformation. Finally, the transformed features are clustered into a foreground (tumor) and a background class. Fig. 2 shows a schematic overview of the lung tumor segmentation approach. The details of the individual stages are discussed in the following.

4.1. Co-registration

We transform the PET image volume from each PET/MRI pair by a B-spline transformation model to align with the MRI scan. The registration is performed unsupervised using the Elastix software (Klein, Staring, Murphy, Viergever, & Pluim, 2009) in Python by running the SimpleElastix toolbox (Marstal, Berendsen, Staring, & Klein, 2016). Elastix is an openly available and frequently used software package for intensity-based medical image registration where the registration problem is formulated as an optimization problem and solved iteratively (Viergever et al., 2016). In this work, the cost function consists of a similarity measure, defined by the mutual information between the two images, and a regularisation term penalizing the displacement magnitude. The cost is optimized in an iterative manner using adaptive stochastic gradient descent in a three-stage pyramidal multi-resolution approach.

Table 1
Detailed information about the dataset and generated supervoxels.

Patients	Total number	18
	Mean age (at exam) [yrs]	72.1
Gender	Male	12
	Female	6
Tumors	Total number	19
Pathology	Adenocarcinoma	12
	Squamous cell carcinoma	7
Resolution	PET [mm]	[2 × 2 × 2]
	MRI [mm]	[1 × 1 × 5]
Ground truth	Median TV (union) [mm^3]	9712
	Median TV (MRI) [mm^3]	6664
	Median TV (PET) [mm^3]	8672
Supervoxels	Median no supervoxels	1495
	Minimum no supervoxels	1480
	Maximum no supervoxels	1511
	Minimum size [mm^3]	1600
	Maximum size [mm^3]	40672

TV = tumor volume.

4.2. Supervoxel generation

The idea of supervoxels is to group similar voxels into basic regions that are more meaningful than individual voxels. In this way, supervoxels capture redundancy in the image and provide local image features.

We apply the simple linear iterative clustering (SLIC) algorithm (Achanta et al., 2010), which, in the field of medical image analysis, has extensively been used as a pre-processing step to reduce the computational cost and the effects of noise and imperfect co-registration (Even et al., 2017; Lucchi, Smith, Achanta, Knott, & Fua, 2011; Roth, Farag, Lu, Turkbey, & Summers, 2015; Soltaninejad et al., 2017).

The SLIC algorithm is based on k-means clustering with k cluster centers initialized on a regular grid with intervals of $S = \sqrt[3]{N/k}$, where N is the number of voxels in the image volume. However, two particular modifications differentiate the SLIC algorithm from standard k-means clustering: (1) Whereas standard k-means searches the entire image, the search space in SLIC is limited to a region proportional to the initial supervoxel size N/k. (2) The distance measure D is a weighted combination of the intensity proximity and the spatial proximity in order to control the size and compactness of the supervoxels (Achanta et al., 2012):

$$D = \sqrt{d_c^2 + \left(\frac{d_s}{S}\right)^2 m^2}, \tag{1}$$

where m is a constant controlling the compactness of the supervoxels, and the distances d_c and d_s, for the case of a three-dimensional grayscale image with voxel intensities l and spatial cooridnates (x, y, z), are given by

$$d_c = \sqrt{(l_j - l_i)^2}, \tag{2}$$

and

$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}. \tag{3}$$

The supervoxel generation is based on the same approach as in Even et al., 2017. All image volumes are z-normalized (subtract mean and divide by standard deviation) and for each image pair $\{I_i^{MRI} I_i^{PET}\}_{i=1}^{N_p}$, an average image volume I_i^z is computed and used to extract supervoxels according to the SLIC algorithm. An initial number of $k = 1500$ supervoxels per patient reduces the total number of data points from 28.7 millions (voxels) to less than 27,000 (supervoxels). The left part of Fig. 3 shows an example slice for one patient.

4.3. Feature extraction

To make the analysis clean, we extract two basic intensity features for each supervoxel, i, and define the feature vector

$$\boldsymbol{x}_i = \left[x_i^{MRI}, x_i^{PET}\right], \tag{4}$$

where x_i^{MRI} is the median intensity within the volume of supervoxel i in the MRI image and x_i^{PET} is the median intensity within the volume of supervoxel i in the PET image (Fig. 3, right). By extracting the median intensities, the effects from outlier voxels are suppressed.

Fig. 4 shows a scatter plot of the extracted feature vectors from all patients, with x^{MRI} on the x-axis and x^{PET} on the y-axis. The colors indicate the fraction of tumor voxels (according to the ground truth

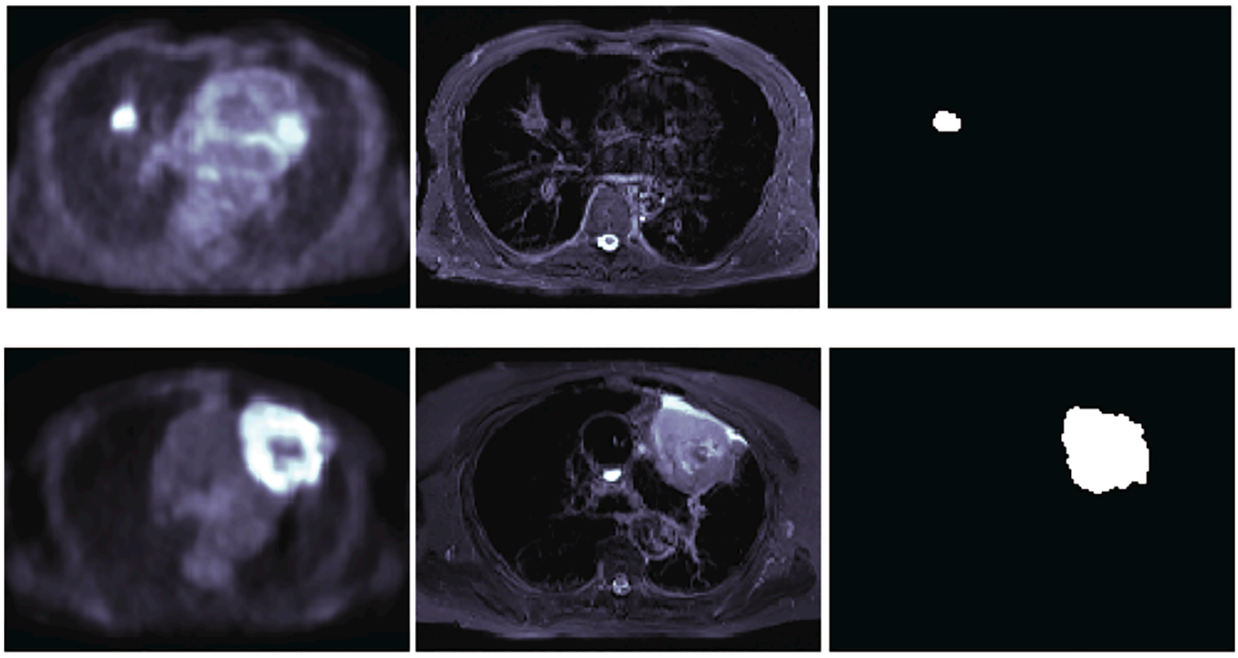


Fig. 1. Two PET/MRI pair examples. Left: PET image slice. Middle: Corresponding MRI image slice. Right: Corresponding ground truth mask indicating the tumor.

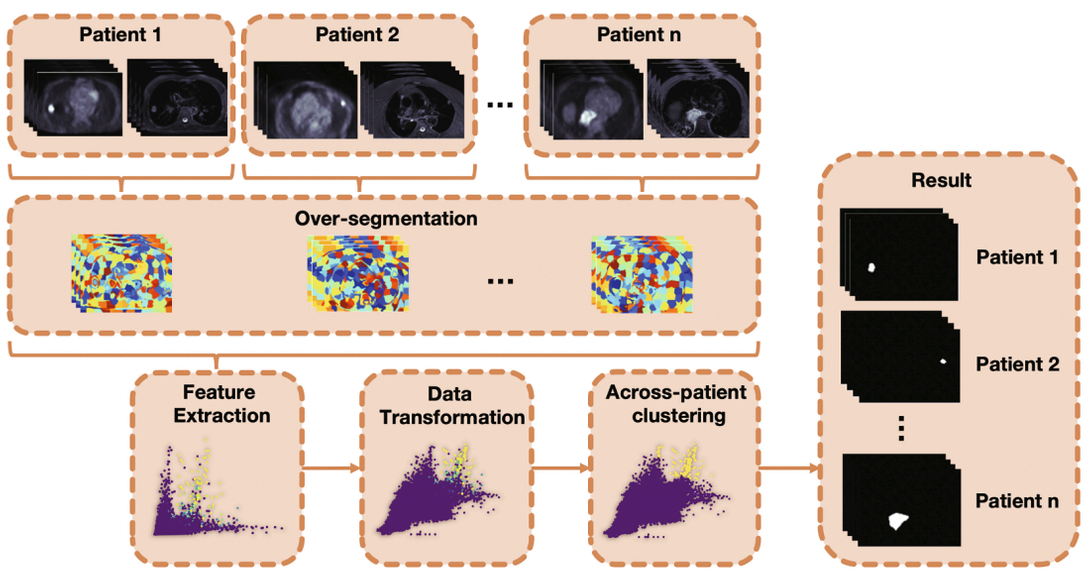


Fig. 2. Schematic of the across-patients supervoxel-based clustering for lung tumor segmentation in hybrid PET/MRI. Based on each co-registered PET/MRI image pair, an over-segmentation is performed to generate supervoxels. From each supervoxel in every patient, two basic intensity features are extracted from the PET/MRI, resulting in a two-dimensional feature space. This feature space is transformed to improve the following clustering into two clusters. Finally, the clustering labels are mapped back to pixel space, giving the resulting segmentation masks. Note that the colors in the scatter plots *before* the clustering indicate the supervoxels' tumor fractions according to the ground truth and is only used for illustration purposes.

labels) within the supervoxels, where yellow translates to pure tumor supervoxel and purple corresponds to pure background supervoxel. This plot illustrates that both modalities contribute with important information in the segmentation task: thresholding any of the two marginal distributions will lead to significant mixing of tumor and non-tumor supervoxels.

4.4. Data transformation

As the original form of the data is not necessarily more suitable for analysis than any function of the data, transformations often play an important role in exploratory data analysis (Stoto & Emerson, 1983). We apply the Box-Cox transformation (Box & Cox, 1964), a long-established power transformation. This transform is a widely used pre-processing step in various fields of applications (Hossain, 2011; Liu, Yin, Wang, & Wang, 2013; Rayens & Srinivasan, 1991; Boroojeni et al., 2017), including tumor detection in MRI (Vos, Barentsz, Karssemeijer, & Huisman, 2012) and classification of lung nodules in CT (Shah et al., 2005). The transformed data $y_i^{(\lambda)}$ is given by

$$y_i^{(\lambda)} = \begin{cases} \dfrac{y_i^{\lambda} - 1}{\lambda}, & \text{if} \lambda \neq 0, \\ \ln y_i, & \text{if} \lambda = 0, \end{cases} \tag{5}$$

where the parameter λ is found by maximizing the log-likelihood under the assumption that the *transformed* data is Gaussian, thereby

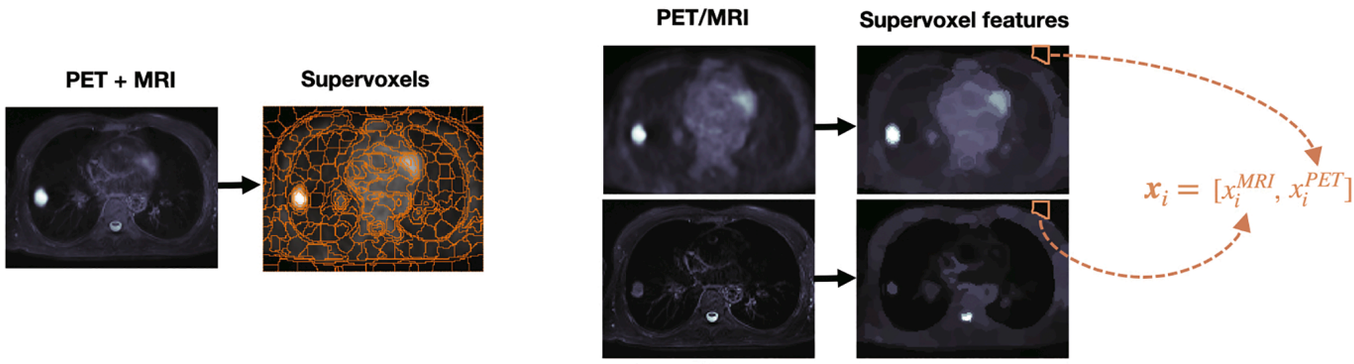


Fig. 3. Illustration of supervoxel generation and feature extraction. Left: From the average image volume I^z, computed based on the z-normalized PET and MRI volumes, we compute the supervoxels. In this specific image slice, the 17,000 voxels are aggregated into 315 supervoxels. Right: Within each supervoxel, we compute the median PET and the median MRI intensity and extract these as supervoxel feature vectors.

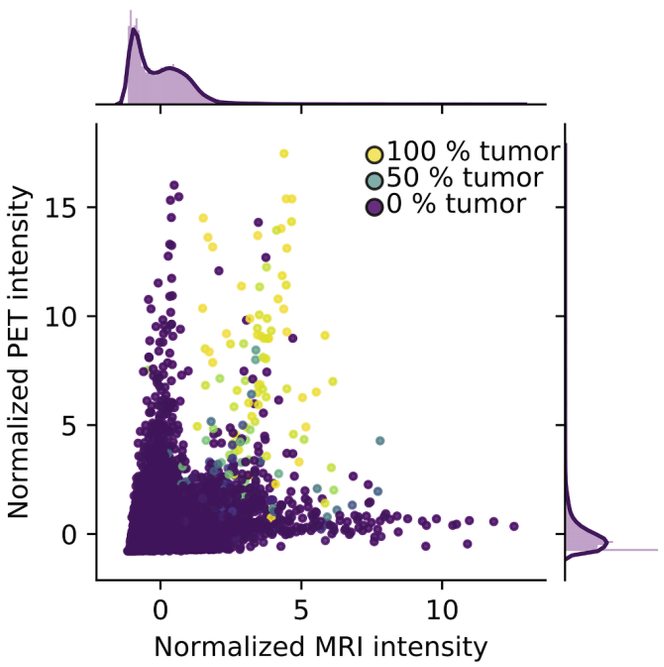


Fig. 4. Scatter plot showing the z-normalized feature space. The x-axis represents the MRI feature whereas the y-axis represents the PET feature.

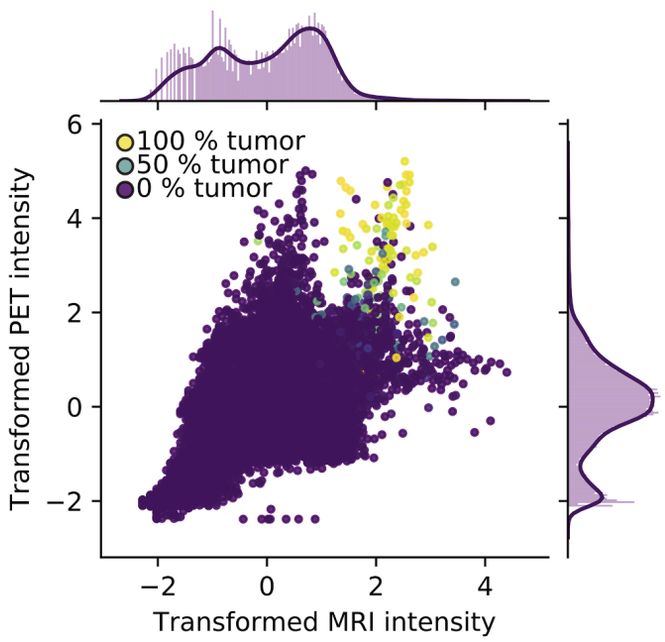


Fig. 5. Scatter plot showing the z-normalized Box-Cox transformed feature space. The x-axis represents the transformed MRI feature whereas the y-axis represents the transformed PET feature.

encouraging the transformed data to be more Gaussian.

Since the SUV values in the original PET images lie in the range [0, 12] whereas the MRI intensities lie in the range [0,1000], we additionally apply a z-normalization to the transformed data. Fig. 5 shows the normalized feature plot after Box-Cox transformation.

4.5. Clustering

Cluster analysis is the study of discovering natural groupings in unlabeled data, such that samples within the same cluster are *similar* and samples in different clusters are *dissimilar*. There exist thousands of clustering algorithms in the literature, and different clustering algorithms (and their parameter settings) often result in different groupings. However, no general "best clustering algorithm" can be named (Jain, 2010). In some way or another, each algorithm enforces a structure on the data and depending on the fit between the model and the data, the resulting clusters will be "good" or "bad" (Jain, 2010).

In this paper, we consider some of the most well-known clustering algorithms in the literature in order to perform lung tumor segmentation in an unsupervised manner. We examine k-means clustering, spectral clustering, and hierarchical clustering. For the benefit of the reader not familiar with these algorithms, we provide a short overview in the following.

4.5.1. K-means clustering

K-means is, due to its simplicity and computational efficiency, one of the most used clustering algorithms in the literature (Jain, 2010). The algorithm partitions the data into k disjoint clusters in a two-step iterative optimization of the cost function, given by (Bishop, 2006):

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\mathbf{x}_n - \boldsymbol{\mu}_k||^2, \tag{6}$$

where N is the number of data points (supervoxels), $\mathbf{x}_n$ is the feature vector of the nth data point, $r_{nk} \in \{0, 1\}$ is the cluster assignment of the nth data point to cluster k, and $\boldsymbol{\mu}_k$ is the cluster representative of the kth cluster, given by the mean of the feature vectors assigned to that cluster.

The algorithm is initialized by choosing a set of initial cluster representatives. Then, each iteration consists of two steps which are repeated until convergence:

1. Assign $\mathbf{x}_n, n = 1, \ldots, N$ to the closest cluster, defined by its cluster representative $\boldsymbol{\mu}_k, k = 1, \ldots, K$.
2. Update cluster representatives $\boldsymbol{\mu}_k, k = 1, \ldots, K$ as the mean of all data points assigned to it.

4.5.2. Hierarchical clustering

Another common clustering approach is hierarchical clustering. In this work we employ hierarchical *agglomerative* clustering, which is the mode where all data points (supervoxels) start out as separate clusters. The algorithm then consists of recursively merging the most similar pair of clusters until we are left with one big cluster, in this way producing a hierarchy of nested clusterings (Theodoridis & Koutroumbas, 2008).

In order to identify the most similar pair of clusters in each iteration, the proximity g between all possible pairs of clusters (C_i, C_j) is computed as a function of the set of affinities between pairs of observations in C_i and C_j (Theodoridis & Koutroumbas, 2008). This requires us to define a measure of proximity between data points (vectors) *and* between clusters (sets of vectors). Thus, depending on the chosen measure of affinity between data points and linkage between clusters, the clustering algorithm may lead to completely different clustering results.

Denoting d_{mn} the *dissimilarity* between observation m in cluster C_i and observation n in cluster C_j, we can define average linkage (Hastie, Tibshirani, & Friedman, 2009):

$$g_{CL}(C_i, C_j) = \frac{1}{N_i N_j} \sum_{m \in C_i} \sum_{n \in C_j} d_{mn}, \tag{7}$$

where N_i and N_j are the number of observations in cluster i and j, respectively.

The average linkage measures the average dissimilarity between the clusters and is a compromise between measuring the dissimilarity between the most similar observations (single link) and the most dissimilar observations (complete link) in different clusters. This middle-ground approach is known to be less sensitive to noise and outliers as the measure is based on all observations in the clusters. As for the dissimilarity between observations d_{mn}, we examine three different measures: the Euclidean norm, the Manhattan norm, and the Cosine distance, leading to three different average linkage clustering algorithms.

4.5.3. Spectral clustering

The third and final clustering approach that we consider in this work is spectral clustering, which has become one of the most used clustering algorithms in recent years (Von Luxburg, 2007). It exploits the spectrum of the affinity matrix to perform clustering and is designed for non-convex problems (Hastie et al., 2009).

In spectral clustering, we represent our data in the form of a similarity graph. Each vertex corresponds to an observation (supervoxel) and the edges connecting pairs of vertices are weighted by their pair-wise similarity. The problem of clustering can then be formulated as a graph-cut problem where we are looking for a graph partitioning such that edges between subsets have low weight and edges within subsets have high weight (Von Luxburg, 2007).

To construct the graph, we first have to decide on a similarity measure, and one of the most common choices is the radial basis function (Theodoridis & Koutroumbas, 2008). This is a Gaussian similarity function which encodes the relation between observations in a local neighborhood. The function is given by

$$a(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2\sigma^2} \right), \tag{8}$$

where σ is a scaling parameter controlling the width of the neighborhood (Von Luxburg, 2007). The affinity matrix $\boldsymbol{A}$ containing the pairwise similarities $a_{ij} = a(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between all n observations can then be used to define the graph Laplacian:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}, \tag{9}$$

where $\boldsymbol{D}$ is the degree matrix, defined as a diagonal matrix with $d_{ii} = \sum_j a_{ij}$. This particular matrix is known to have an important property that can be used to change the representation of the data (Von Luxburg, 2007): For every vector $\boldsymbol{f} \in \mathbb{R}^n$ we know that

$$\boldsymbol{f}' \boldsymbol{L} \boldsymbol{f} = \frac{1}{2} \sum_{i,j=1}^{n} a_{ij} (f_i - f_j)^2. \tag{10}$$

Eq. (10) can be seen as the eigenvalue decomposition of $\boldsymbol{L}$,

$$\boldsymbol{f}' \boldsymbol{L} \boldsymbol{f} = \lambda, \tag{11}$$

which means that the eigenvector $\boldsymbol{f}$ can be thought of as a fuzzy indication vector, indicating a partitioning of the graph resulting in a cut cost corresponding to its eigenvalue λ. Spectral clustering exploits this result by containing the m eigenvectors of $\boldsymbol{L}$ corresponding to the m smallest eigenvalues in a matrix $\boldsymbol{F}_m \in \mathbb{R}^{n \times m}$ and performs k-means clustering on its rows.

In this paper, we employ spectral clustering with the normalized graph Laplacian, defined by

$$\boldsymbol{L}_n = \boldsymbol{D}^{-1/2} \boldsymbol{L} \boldsymbol{D}^{-1/2}. \tag{12}$$

This matrix has properties similar to $\boldsymbol{L}$ and is usually preferred for reasons discussed by Von Luxburg, 2007.

5. Experiments and results

In this section, we evaluate the above-mentioned clustering algorithms on the task of lung tumor segmentation. We seek a clustering consisting of two clusters (tumor and non-tumor supervoxels) and we analyze the performance of the different clustering algorithms, as well as the influence of the proposed pre-processing steps. That is, we apply the clustering algorithms to z-normalized Box-Cox transformed data (referred to as transformed data) and evaluate the results quantitatively and qualitatively. For ease of comparison, all quantitative results are summarized in Table 2.

5.1. Evaluation measure

To quantitatively compare the segmentation performance of the clustering methods with the manual delineations, we use the voxel-wise dice score. The dice score, D, between two segmentations A and B is given by

$$D(A, B) = 2 \frac{|A \cap B|}{|A| + |B|}, \tag{13}$$

which means that a dice score of 1 corresponds to a perfect match between the segmentations.

We compute both *overall* dice score (treating the labels of all patients as one segmentation), and *patient-wise* dice score where we report the mean dice score and the standard deviation over all patients.

Table 2
Quantitative results of 2-class clustering on the Box-Cox transformed data.

Method	Mean	SD	OA	#Miss
K-means	0.011	0.015	0.015	8/19
Hierarchical_E	0.288	0.294	0.361	4/19
Hierarchical_M	0.461	0.321	0.657	5/19
Hierarchical_C	0.013	0.017	0.013	7/19
Spectral	0.470	0.308	0.668	5/19

Mean, standard deviation (SD) and overall (OA) dice score. #Miss is the number of tumors completely missed in the segmentation.

5.2. K-means clustering

As is apparent from Table 2, k-means clustering into two clusters leads to poor performance with respect to dice score and we completely miss 8 out of 19 tumors. Taking a closer look at the clustering result by mapping the labels back to the image domain, we see in Fig. 6 that the clusters roughly represent "air/lung" and "tissue/bone/tumor" and not "tumor", "non-tumor". This is not uncommon in the unsupervised setting, where the model is not steered to produce specific classes.

In order to further analyze the performance of k-means clustering, we successively increase the number of desired clusters up to $k = 30$ and determine the best possible performance that can be achieved in each step: If the one "best cluster" out of the produced number of clusters is selected to represent "tumor" and the union of the remaining clusters is treated as "non-tumor", we can compute the maximum dice score for each configuration, shown by the blue curve in Fig. 7. As is apparent from the plot, the dice score increases as the number of clusters increases and we can achieve dice scores higher than 0.7 if we use a high enough k. In practice, the selection of the "best cluster" could be performed by medical experts, but we resort to finding the cluster that gives the best performance using label information (the label information is only used for evaluation). However, the merging of all non-tumor clusters is a non-trivial task and is not feasible in practice.

To examine the effect of the Box-Cox transform, we have also included the results of clustering the non-transformed data (orange curve) in Fig. 7. This curve converges at a lower dice score, which is related to k-means' known problems with elongated clusters and tendency to cluster the data into compact and uniform sized clusters. This experiment further confirms our suspicion that the Box-Cox transformation improves clusterability and we, therefore, consider only the transformed data in the remaining experiments.

5.3. Hierarchical clustering

Table 2 presents the results of clustering transformed data into two clusters using the different hierarchical clustering algorithms. We see that clustering with Manhattan distance measure achieves the highest overall (0.657) and mean (0.461 ± 0.321) dice score. Further, we see that the Euclidean distance measure achieves significantly lower dice scores, but misses only four out of 19 tumors, whereas the Cosine distance measure yields low dice scores and a high number of missed tumors.

Fig. 8 shows the clustering results mapped back to the image domain for five tumor slices in five different patients. It is apparent that both the Euclidean and Manhattan distance measure seem to cluster the data roughly into "tumor" and "non-tumor", whereas the Cosine distance measure leads to a poor segmentation performance, similar to k-means.

5.4. Spectral clustering

In spectral clustering, the affinity matrix is computed using the radial basis function, which is standard practice (Theodoridis & Koutroumbas, 2008). As the scaling parameter σ decides the width of the neighborhoods in which we encode the relations between observations, the parameter choice is critical for the clustering result. Here, we apply the rule of thumb given in Jenssen, 2009:

$$\sigma = 0.15 \cdot \text{median}\{d_{ij}\}_{i,j=1}^{n}, \tag{14}$$

where d_{ij} is the Euclidean distance between feature vector i and j. Nonetheless, in our experiments, we observed that the results are robust to the choice of σ. Since the radial basis function results in a connected graph, the eigenvector corresponding to the smallest eigenvalue (=1) is constant (Von Luxburg, 2007). We, therefore, ignore the smallest one and look at the subsequent eigenvectors. Fig. 9 shows a plot of the 2nd, 3rd, and 4th smallest eigenvectors mapped back to the image domain for one slice in five different patients. From the first row, we see that the cheapest cut (2nd smallest eigenvector) corresponds to (soft) partitioning the graph into two subsets roughly representing "air surrounding the patient" and "patient". Moving on to the third eigenvector (second row in Fig. 9), we see that it appears to detect the tumors. The most common approach in spectral clustering is to use as many eigenvectors in the final k-means step as there are classes in the data. However, as the third eigenvector seems to have the most information about the tumors, we first cluster the data into two clusters based on this eigenvector alone. This yields an overall dice score of 0.668 and a mean dice of 0.470 ± 0.308. No improvements were observed when including the second and fourth eigenvectors.

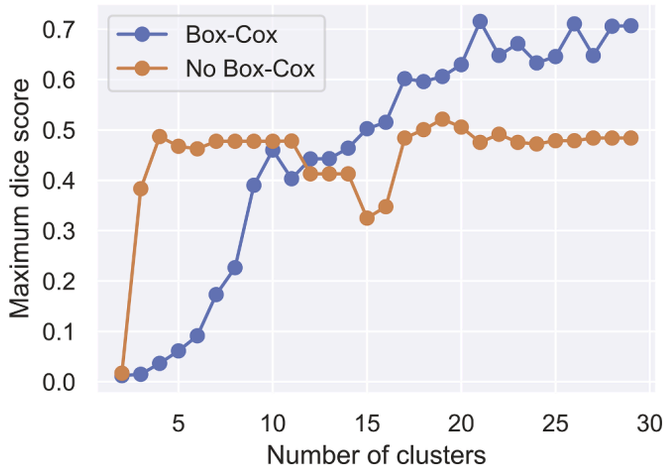


Fig. 7. K-means as function of number of clusters. Performance of k-means clustering with respect to maximum overall (OA) dice score as a function of number of clusters for standard scaled data (orange) and Box-Cox transformed data (blue). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5.5. Effect of tumor size

In our data set we have 19 tumors ranging in size from 1944 mm^3 to

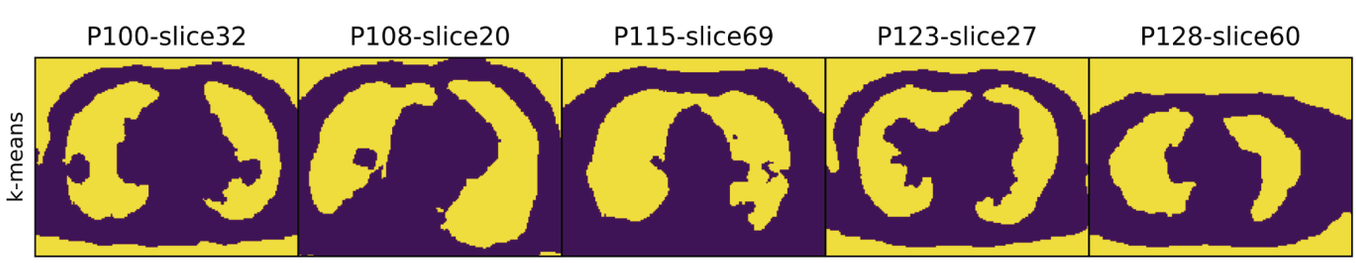


Fig. 6. Visualization of k-means clustering. Result of mapping the k-means $k = 2$ clustering labels back to the image domain and displaying the segmentations for five tumor containing slices in five different patients. The two clusters roughly represent "air/lung" (yellow) and "tissue/bone/tumor" (purple). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

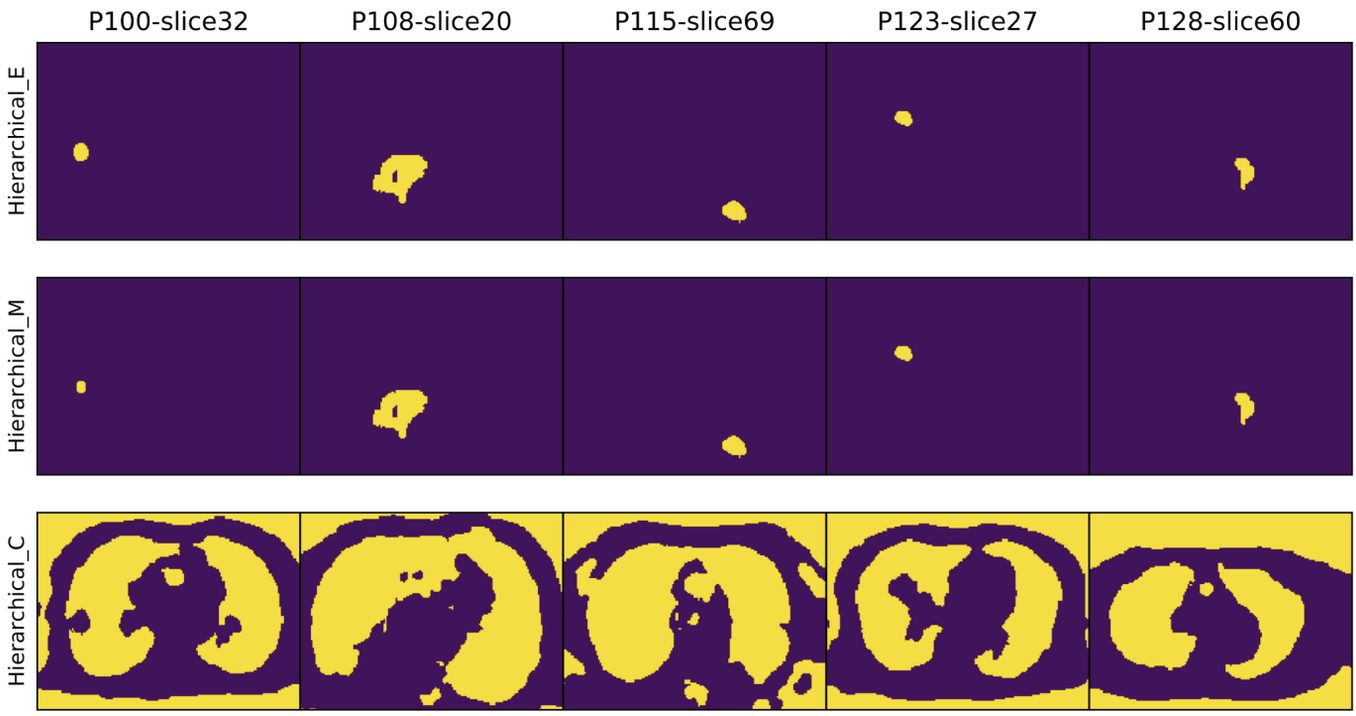


Fig. 8. Visualization of hierarchical clustering. Result of mapping the Euclidean (top), Manhattan (middle) and Cosine (bottom) hierarchical clustering labels back to the image domain and displaying the segmentations for five tumor-containing slices in five different patients. For Euclidean and Manhattan hierarchical clustering, the two clusters roughly represent tumor (yellow) and non-tumor (purple). For Cosine hierarchical clustering, the clusters roughly represent "air/lung" (yellow) and "tissue/bone/tumor" (purple). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

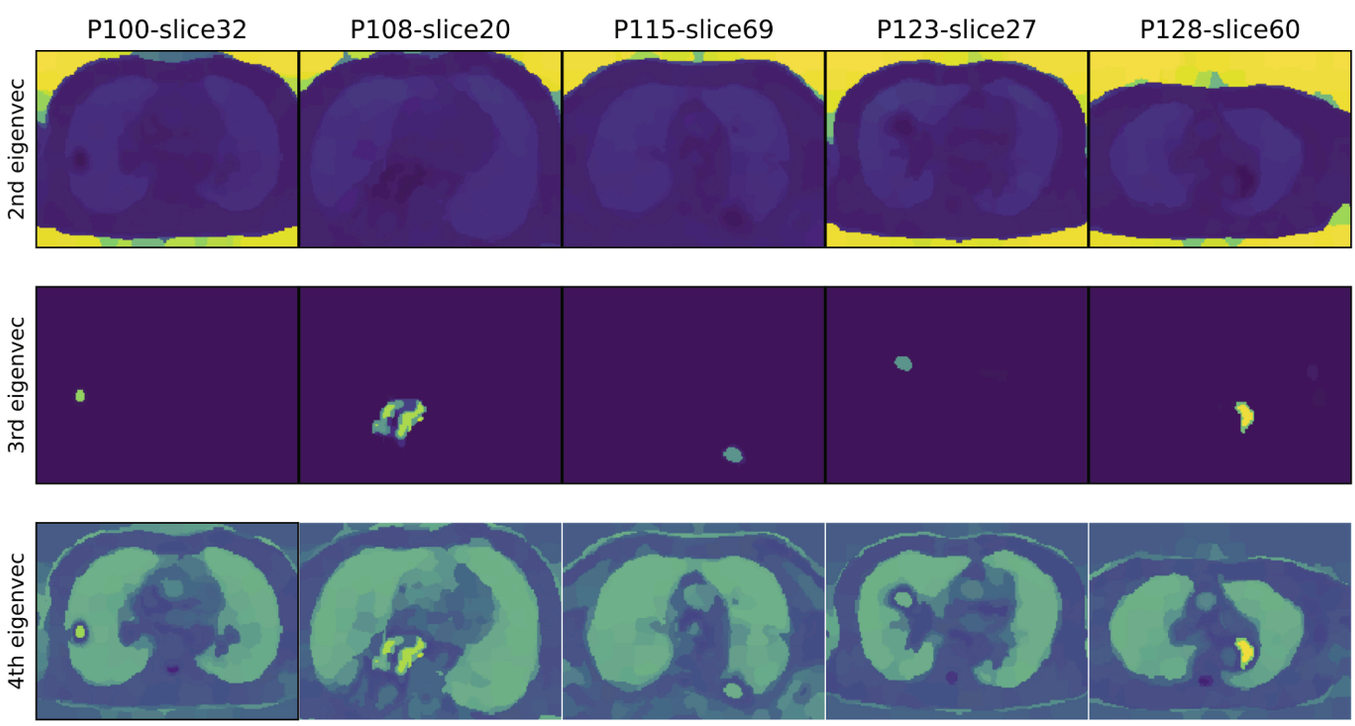


Fig. 9. Visualization of eigenvectors. In spectral clustering, the eigenvectors of the Laplacian can be thought of as fuzzy indication vectors, indicating a partitioning of the graph resulting in a cut cost corresponding to their eigenvalues. This figure shows the eigenvectors corresponding to the 2nd, 3rd and 4th smallest eigenvalue mapped back to pixel space in five tumor slices for five different patients. The 2nd smallest eigenvector (first row) seems to partition the graph into two subsets roughly representing "air surrounding the patient" and "patient". The third eigenvector (second row) appears to pick up on the tumors whereas the fourth eigenvector does not contain additional information about the tumors. Note that we exclude the first eigenvector as it is constant for a connected graph.

$195,744$ mm^3. In order to analyze the effect of tumor size on the clustering dice, we define two thresholds that divide the tumors into three groups; eight small-sized tumor ($< 8000\text{mm}^3$), six medium-sized tumors ($\in [8000, 80,000]\text{mm}^3$) and five large-sized tumors ($> 80,000\text{mm}^3$). The box-plot in Fig. 10 presents the segmentation performance with respect to dice score for these three groups. Note that we omitted k-means

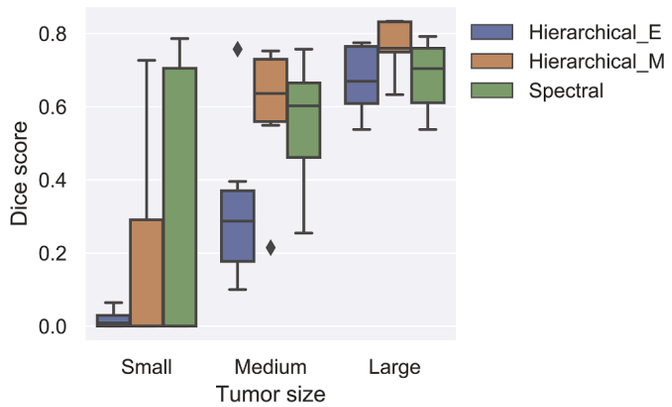


Fig. 10. Effect of tumor size. Grouped box-plot showing the effect of tumor size on the segmentation dice score for Euclidean hierarchical (blue), Manhattan hierarchical (orange) and spectral (green) clustering. The tumors are divided into three groups; small (left), medium (middle) and large (right) size tumors. The mean dice score generally increases while the variance decreases for larger tumors. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

clustering and hierarchical clustering with Cosine distance in this comparison as they did not manage to pick up on the tumors. As can be seen in the box-plot, there is a trend towards better dice score with larger tumor size for all methods. The mean dice score generally increases while the variance decreases, yielding more robust predictions for larger tumors. Specifically, we see that none of the algorithms provide reliably high scores for small tumors, but that spectral clustering is able to capture *some* of the small tumors with good dice scores. Further, the difference in segmentation performance (with respect to dice score) among the algorithms decreases with increasing tumor size. Euclidean hierarchical clustering, for instance, gets a dice score close to zero for small tumors but seems to perform equally good as spectral clustering for the large tumors.

5.6. Analysis of segmentation errors

As the dice score treats false negatives and false positives equally, it is also important to evaluate the types of mistakes that each clustering algorithm makes. The bar plot in Fig. 11 presents the number of true positive (TP), false negative (FN) and false positive (FP) voxels in each of the segmentations obtained from the different clustering algorithms. The most interesting result from this analysis is that spectral clustering, which overall achieves the highest dice score, turns out to have the highest number of FNs *and* the lowest number of TPs, providing an overly optimistic segmentation (under-estimation of the tumor volume). Manhattan hierarchical clustering, on the other hand, which, according

Fig. 11. Segmentation errors. Bar plot presenting the number of TP: true positive (tumor) voxels; FN: false negative (non-tumor) voxels; FP: false positive voxels. The dotted line indicates the total number of TP tumor voxels.

to the dice score has comparable performance, shows to actually have the highest number of TPs and lowest number of FNs, in this way being the method detecting most tumor voxels. Euclidean hierarchical clustering, which has the lowest dice score of the three methods to be compared, has a comparable (to Manhattan hierarchical clustering) number of TPs and FNs, but has too many FPs, resulting in an overly pessimistic segmentation. This means that even though the Euclidean hierarchical clustering fails completely for small tumors according to the dice score, it does *not* necessarily miss the tumors in the segmentation. Note that the sum of the segmentation mask and the ground truth mask in the denominator in Eq. 13 makes the dice score more sensitive to over-segmentation of small tumors compared to larger tumors.

Fig. 12 shows the effects of the different types of mistakes in the image domain. In general, we see that Euclidean hierarchical clustering does not miss the tumors, but tends to over-segment the tumor class by including other organs, resulting in a large number of FPs. Spectral clustering, on the other hand, tends to under-segment the tumor volume, whereas Manhattan hierarchical clustering captures the most tumor voxels without having an alarmingly high number of FP. The two rightmost columns in Fig. 12 show two slices from the same patient at different positions and further illustrate the over-segmentation issue of the hierarchical algorithms (note that slice 5 does not contain tumor voxels).

An interesting observation in P118-slice50 in Fig. 12 is that there is a "hole" in the tumor, which is a common phenomenon for large tumors in PET imaging. The apparent "hole" is most likely caused by necrosis, occurring due to shortage of oxygen supply to the tumor. We see that the Manhattan hierarchical clustering succeeds in exploiting the combined information from both modalities and provides a closed segmentation, as desired.

Regarding the complete misses, we find that Euclidean hierarchical clustering completely misses four out of the nineteen tumors, where two of the misses come from the same patient. Spectral clustering and Manhattan hierarchical clustering miss the same four tumors, in addition to one more (the same one for both). By inspecting the number of overlapping tumor voxels between the two modalities (using the ground truth segmentation masks), we find that the five tumors that are completely missed by the algorithms are among the six tumors with the least number of overlapping tumor voxels across modalities. Moreover, we find that for two of the missed tumors, the maximum SUV within the ground truth segmentation is lower than 1.3, which is a particular low uptake value in PET.

In a clinical setting, the detection of tumors is arguably of utmost importance and over-segmentation is preferred. Only focusing on the number of tumors that are missed completely and therefore choosing the Euclidean hierarchical clustering, would, however, result in a large number of false-positive voxels. This means that the clinicians would be presented with many potential tumors that they have to evaluate, which in turn could lead to real tumors getting missed. Spectral clustering and Manhattan hierarchical clustering, on the other hand, achieve a low number of misses, while at the same time producing a low number of false-positive voxels.

5.7. Benefit of clustering across patients

An important contribution of this paper is the across-patients clustering to improve segmentation performance. In order to quantify the benefit of clustering across patients, we conduct an experiment where we Box-Cox-transform and cluster the supervoxel features of each patient separately. Fig. 13, illustrates the results and visualizes the difference in performance when considering patient-by-patient clustering versus across-patients clustering for the two best performing models. Across-patients clustering achieves considerable improvements for most tumors. For spectral clustering, for instance, the mean dice score increases from 0.169 ± 0.295 patient-by-patient to 0.470 ± 0.308 across-patients. The results for the individual patients can also be found in

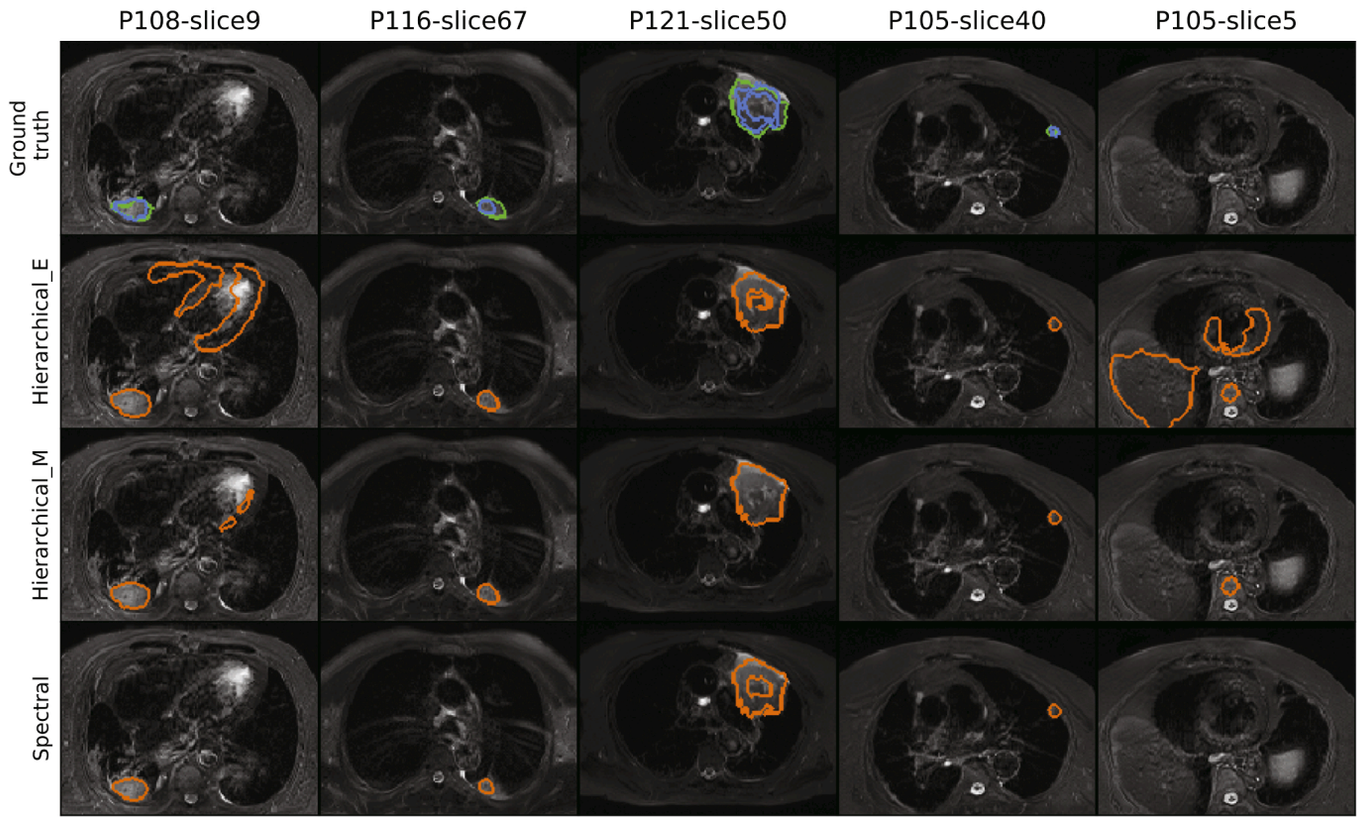


Fig. 12. Segmentation errors in image domain. Overlaying segmentation results (orange) on the MRI for a few selected slices for the different clustering algorithms. The first row shows the ground truth (PET delineation in blue and MRI delineation in green). Note that columns 4–5 show two slices from the same patient at different positions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

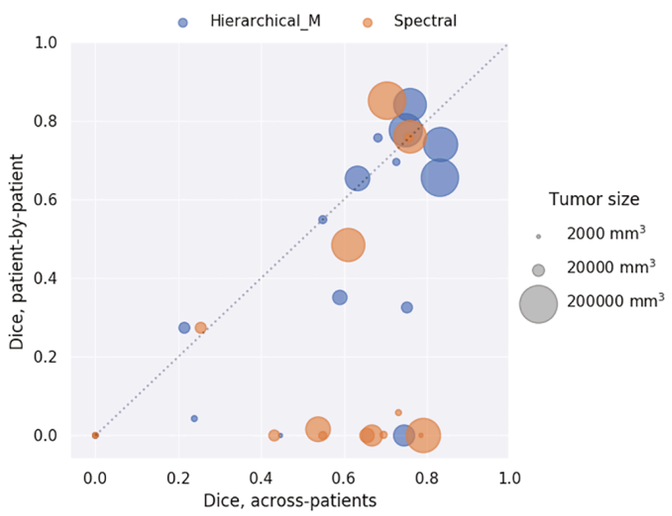


Fig. 13. Patient-by-patient clustering versus across-patients clustering. Scatter plot showing dice score for patient-by-patient clustering against dice score for across-patients clustering. For tumors under the diagonal, the across-patients clustering achieved better dice scores than the patient-by-patient clustering did.

Table 3.

Table 3
Patient-by-patient versus across-patients clustering. Dice scores for patient-by-patient (P-by-p) and across-patients (Across-p) clustering for the two best performing models.

	Dice score			
	Spectral		Hierarchical_M	
Patient	P-by-p	Across-p	P-by-p	Across-p
100	0.0	0.549	0.550	0.549
102	0.0	0.0	0.0	0.0
103a	0.0	0.0	0.0	0.0
103b	0.0	0.0	0.0	0.0
104	0.0	0.786	0.0	0.447
105	0.0	0.656	0.351	0.590
108	0.485	0.612	0.777	0.750
109	0.0	0.0	0.0	0.0
112	0.016	0.538	0.654	0.633
114	0.274	0.254	0.274	0.215
115	0.0	0.432	0.326	0.752
116	0.760	0.760	0.841	0.760
118	0.852	0.704	0.656	0.832
121	0.0	0.0	0.0	0.0
123	0.058	0.732	0.043	0.239
125	0.0	0.668	0.0	0.745
128	0.757	0.757	0.757	0.682
129	0.0	0.792	0.740	0.834
131	0.002	0.696	0.696	0.727
Mean	0.169	0.470	0.351	0.461
Std	0.295	0.308	0.328	0.321

5.8. Noise robustness

Differences in imaging protocols and acquisition conditions can result in variations in the signal-to-nose-ratio. To evaluate the noise robustness of the two best performing methods, we therefore simulate a reduced signal-to-noise-ratio by adding noise to the images before performing the segmentation. Following Jayender, Chikarmane, Jolesz, and Gombos, 2014, we add white Gaussian noise with standard deviation equal to 5%, 15% and 25% of the base intensity of each voxel in the PET and MRI images. Fig. 14 shows an example slice in PET (top) and

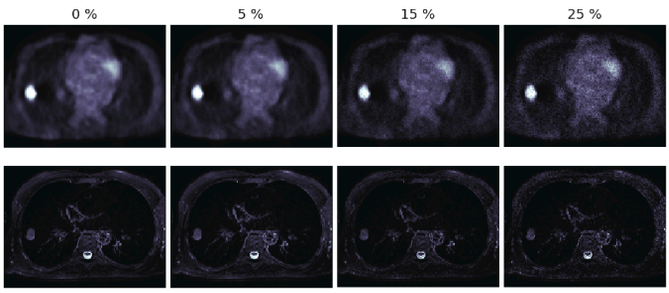


Fig. 14. Noise analysis. Example slices in PET (top) and MRI (bottom) corrupted with white Gaussian noise with standard deviation equal to 5%, 15% and 25% of the voxel intensities.

MRI (bottom) with increasing noise level towards right.

The segmentation maps were computed in the same way as for the original PET/MRI images and compared to the ground truth. The experiments were repeated ten times (with different random seeds in the noise generation) and the results are reported in Fig. 15. Overall, we see that the spectral clustering seems to be robust to 5% and 15% noise, whereas it becomes unstable for 25% noise, with a drop in mean dice and a considerable increase in standard deviation. For Manhattan hierarchical clustering on the other hand, we see that the mean dice score drops significantly and that the standard deviation is high for all noise levels.

6. Outlook and limitations

From our results, we can see that the algorithms detect most of the tumors, but that there still is a relatively high number of tumor voxels that are wrongly segmented. From Fig. 5, it is evident that it is impossible to perfectly cluster tumor voxels and non-tumor voxels into two separate clusters, and the reason for this is twofold. Firstly, some of the supervoxels contain both tumor and non-tumor voxels, and secondly, the chosen features are not able to completely separate tumor-containing and non-tumor-containing supervoxels.

The purity of the supervoxels could in theory be enhanced by increasing the number of supervoxels. However, this comes at the cost of increased computational complexity. In our experiments, a number of 1500 supervoxels per patient was chosen as a middle-ground between supervoxel purity and computational cost. Nevertheless, we can not guarantee that this is the best setting and improved supervoxel generation is left for future work.

The median intensities within the supervoxels are in some cases insufficient to detect a supervoxel as "tumor supervoxel". Other features, such as shape, texture, and histogram features may be able to help the discrimination. Radiomics is a process that extracts large amounts of these types of quantitative image features from medical images and has shown potential to improve tumor classification (Wu et al., 2016). However, how to exploit these large amounts of features in an unsupervised manner is challenging because the variance in the features does not necessarily reflect the classes of interest (tumor and non-tumor). Future efforts should focus on searching for alternative features to improve the discrimination between tumor and non-tumor supervoxels.

Further, there are potential limitations connected to the nature of the data acquisition. Firstly, because of respiratory motion, the PET and MRI can not be assumed perfectly co-registered. An unsupervised co-registration was performed to improve the tumor overlap, but missmatches are still present in the data set. Another limitation is the relatively small sample size.

7. Conclusion

In this paper, we proposed a framework for across-patients supervoxel-based unsupervised lung tumor segmentation in PET/MRI. We

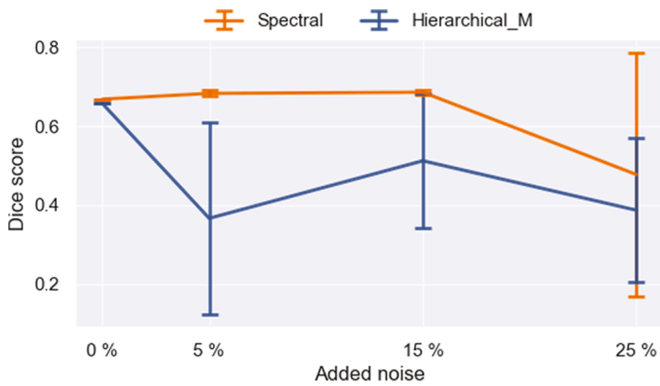


Fig. 15. Noise robustness. Line plot with error-bars showing the effect on the dice score with increasing noise levels for Manhattan hierarchical (blue) and spectral (orange) clustering. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

analyzed the segmentation results for several commonly used clustering algorithms within the framework, investigating their advantages and shortcomings. Results demonstrate that spectral clustering and Manhattan hierarchical clustering have the potential to segment tumors in PET/MRI by producing a low number of missed tumors while maintaining a low number of false-positives. In the presence of low to moderate noise levels, spectral clustering provides stable results whereas Manhattan hierarchical clustering seems to be more sensitive to perturbations in the voxel intensities. The results further highlight the importance of performing clustering across patients, and an analysis of the clustering errors illustrates that it is a particular challenge to segment small-size tumors in the presence of imperfect co-registration. Moreover, the framework represents a step towards generic unsupervised tumor segmentation also beyond the lung tumor segmentation task.

CRediT authorship contribution statement

Stine Hansen: Conceptualization, Software, Methodology, Formal analysis, Visualization, Writing - original draft, Writing - review & editing. **Samuel Kuttner:** Conceptualization, Methodology, Data curation, Writing - review & editing. **Michael Kampffmeyer:** Conceptualization, Methodology, Writing - review & editing, Writing - original draft. **Tom-Vegard Markussen:** Data curation, Writing - review & editing. **Rune Sundset:** Validation, Writing - review & editing. **Silje Kjærnes Øen:** Data curation, Resources, Writing - review & editing. **Live Eikenes:** Data curation, Resources, Writing - review & editing. **Robert Jenssen:** Conceptualization, Methodology, Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank Karl Øyvind Mikalsen and Stian Normann Anfinsen for insightful discussions. This work is supported by the Northern Norway Regional Health Authority (Grant No. HNF1349-17), the Central Norway Regional Health Authority (Grant No. 46056912), and the Norwegian Research Council (Grant No. 303514)

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2010). Slic superpixels. Technical Report.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34*, 2274–2282.

Bagci, U., Udupa, J. K., Mendhiratta, N., Foster, B., Xu, Z., Yao, J., Chen, X., & Mollura, D. J. (2013). Joint segmentation of anatomical and functional images: Applications in quantification of lesions from pet, pet-ct, mri-pet, and mri-pet-ct images. *Medical Image Analysis, 17*, 929–945.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Boroojeni, K. G., Amini, M. H., Bahrami, S., Iyengar, S., Sarwat, A. I., & Karabasoglu, O. (2017). A novel multi-time-scale modeling for electric power demand forecasting: From short-term to medium-term horizon. *Electric Power Systems Research, 142*, 58–73.

Box, G. E., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological), 26*, 211–243.

Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., & Jemal, A. (2018). Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians, 68*, 394–424.

Caldwell, C. B., Mah, K., Ung, Y. C., Danjoux, C. E., Balogh, J. M., Ganguli, S. N., & Ehrlich, L. E. (2001). Observer variation in contouring gross tumor volume in patients with poorly defined non-small-cell lung tumors on ct: the impact of 18fdg-hybrid pet fusion. *International Journal of Radiation Oncology* Biology* Physics, 51*, 923–931.

De Bruijne, M. (2016). Machine learning approaches in medical image analysis: From detection to diagnosis.

Ehman, E. C., Johnson, G. B., Villanueva-Meyer, J. E., Cha, S., Leynes, A. P., Larson, P. E. Z., & Hope, T. A. (2017). Pet/mri: where might it replace pet/ct? *Journal of Magnetic Resonance Imaging, 46*, 1247–1262.

Even, A. J., Reymen, B., La Fontaine, M. D., Das, M., Mottaghy, F. M., Belderbos, J. S., De Ruysscher, D., Lambin, P., & van Elmpt, W. (2017). Clustering of multi-parametric functional imaging to identify high-risk subvolumes in non-small cell lung cancer. *Radiotherapy and Oncology, 125*, 379–384.

Flechsig, P., Mehndiratta, A., Haberkorn, U., Kratochwil, C., & Giesel, F. L. (2015). Pet/mri and pet/ct in lung lesions and thoracic malignancies. In *Seminars in nuclear medicine* (pp. 268–281). Elsevier. Vol. 45.

Foster, B., Bagci, U., Mansoor, A., Xu, Z., & Mollura, D. J. (2014). A review on segmentation of positron emission tomography images. *Computers in Biology and Medicine, 50*, 76–96.

Gordillo, N., Montseny, E., & Sobrevilla, P. (2013). State of the art survey on mri brain tumor segmentation. *Magnetic Resonance Imaging, 31*, 1426–1438.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Hossain, M. Z. (2011). The use of box-cox transformation technique in economic and statistical analyses. *Journal of Emerging Trends in Economics and Management Sciences, 2*, 32–39.

Hurkmans, C. W., Borger, J. H., Pieters, B. R., Russell, N. S., Jansen, E. P., & Mijnheer, B. J. (2001). Variability in target volume delineation on ct scans of the breast. *International Journal of Radiation Oncology* Biology* Physics, 50*, 1366–1372.

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters, 31*, 651–666.

Jayender, J., Chikarmane, S., Jolesz, F. A., & Gombos, E. (2014). Automatic segmentation of invasive breast carcinomas from dynamic contrast-enhanced mri using time series analysis. *Journal of Magnetic Resonance Imaging, 40*, 467–475.

Jenssen, R. (2009). Kernel entropy component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*, 847–860.

Ju, W., Xiang, D., Zhang, B., Wang, L., Kopriva, I., & Chen, X. (2015). Random walk and graph cut for co-segmentation of lung tumor on pet-ct images. *IEEE Transactions on Image Processing, 24*, 5854–5867.

Klein, S., Staring, M., Murphy, K., Viergever, M. A., & Pluim, J. P. (2009). Elastix: a toolbox for intensity-based medical image registration. *IEEE Transactions on Medical Imaging, 29*, 196–205.

Kuttner, S., Lassen, M. L., Øen, S. K., Sundset, R., Beyer, T., & Eikenes, L. (2020). Quantitative pet/mr imaging of lung cancer in the presence of artifacts in the mr-based attenuation correction maps. *Acta Radiologica, 61*, 11–20.

Leibfarth, S., Eckert, F., Welz, S., Siegel, C., Schmidt, H., Schwenzer, N., Zips, D., & Thorwarth, D. (2015). Automatic delineation of tumor volumes by co-segmentation of combined pet/mr data. *Physics in Medicine & Biology, 60*, 5399.

Liu, C.-L., Yin, F., Wang, D.-H., & Wang, Q.-F. (2013). Online and offline handwritten chinese character recognition: Benchmarking on new databases. *Pattern Recognition, 46*, 155–162.

Lucchi, A., Smith, K., Achanta, R., Knott, G., & Fua, P. (2011). Supervoxel-based segmentation of mitochondria in em image stacks with learned shape features. *IEEE Transactions on Medical Imaging, 31*, 474–486.

Marstal, K., Berendsen, F., Staring, M., & Klein, S. (2016). Simpleelastix: A user-friendly, multi-lingual library for medical image registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 134–142).

Moghbel, M., Mashohor, S., Mahmud, R., & Saripan, M. I. B. (2018). Review of liver segmentation and computer assisted detection/diagnosis methods in computed tomography. *Artificial Intelligence Review, 50*, 497–537.

Raponi, M., Zhang, Y., Yu, J., Chen, G., Lee, G., Taylor, J. M., MacDonald, J., Thomas, D., Moskaluk, C., Wang, Y., et al. (2006). Gene expression signatures for predicting prognosis of squamous cell and adenocarcinomas of the lung. *Cancer Research, 66*, 7466–7472.

Rayens, W. S., & Srinivasan, C. (1991). Box–cox transformations in the analysis of compositional data. *Journal of Chemometrics, 5*, 227–239.

Roth, H. R., Farag, A., Lu, L., Turkbey, E. B., & Summers, R. M. (2015). Deep convolutional networks for pancreas segmentation in ct imaging. In Medical imaging 2015: Image processing. Vol. 9413. International Society for Optics and Photonics. p. 94131G.

Sauwen, N., Acou, M., Van Cauter, S., Sima, D., Veraart, J., Maes, F., Himmelreich, U., Achten, E., & Van Huffel, S. (2016). Comparison of unsupervised classification methods for brain tumor segmentation using multi-parametric mri. *NeuroImage: Clinical, 12*, 753–764.

Sbei, A., ElBedoui, K., Barhoumi, W., Maksud, P., & Maktouf, C. (2017). Hybrid pet/mri co-segmentation based on joint fuzzy connectedness and graph cut. *Computer Methods and Programs in Biomedicine, 149*, 29–41.

Sbei, A., ElBedoui, K., Barhoumi, W., & Maktouf, C. (2020). Gradient-based generation of intermediate images for heterogeneous tumor segmentation within hybrid pet/mri scans. *Computers in Biology and Medicine, 119*, Article 103669.

Shah, S. K., McNitt-Gray, M. F., Rogers, S. R., Goldin, J. G., Suh, R. D., Sayre, J. W., Petkovska, I., Kim, H. J., & Aberle, D. R. (2005). Computer aided characterization of the solitary pulmonary nodule using volumetric and contrast enhancement features1. *Academic Radiology, 12*, 1310–1319.

Soltaninejad, M., Yang, G., Lambrou, T., Allinson, N., Jones, T. L., Barrick, T. R., Howe, F. A., & Ye, X. (2017). Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in flair mri. *International Journal of Computer Assisted Radiology and Surgery, 12*, 183–203.

Stoto, M. A., & Emerson, J. D. (1983). Power transformations for data analysis. *Sociological Methodology, 14*, 126–168.

Theodoridis, S., & Koutroumbas, K. (2008). *Pattern recognition* (4th ed.). USA: Academic Press Inc.

Viergever, M. A., Maintz, J. A., Klein, S., Murphy, K., Staring, M., & Pluim, J. P. (2016). A survey of medical image registration – Under review.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing, 17*, 395–416.

Vos, P., Barentsz, J., Karssemeijer, N., & Huisman, H. (2012). Automatic computer-aided detection of prostate cancer based on multiparametric magnetic resonance image analysis. *Physics in Medicine & Biology, 57*, 1527.

Wadhwa, A., Bhardwaj, A., & Verma, V. S. (2019). A review on brain tumor segmentation of mri images. *Magnetic Resonance Imaging, 61*, 247–259.

Wu, J., Gensheimer, M. F., Dong, X., Rubin, D. L., Napel, S., Diehn, M., Loo, B. W., Jr, & Li, R. (2016). Robust intratumor partitioning to identify high-risk subregions in lung cancer: a pilot study. *International Journal of Radiation Oncology* Biology* Physics, 95*, 1504–1512.

Wu, W., Parmar, C., Grossmann, P., Quackenbush, J., Lambin, P., Bussink, J., Mak, R., & Aerts, H. J. (2016). Exploratory study to identify radiomics classifiers for lung cancer histology. *Frontiers in Oncology, 6*, 71.

Xu, Z., Bagci, U., Udupa, J. K., & Mollura, D. J. (2015). Fuzzy connectedness image co-segmentation for hybridpet/mri and pet/ct scans. In *Computational Methods for Molecular Imaging* (pp. 15–24). Springer.

Paper II

Anomaly Detection-Inspired Few-Shot Medical Image Segmentation Through Self-Supervision With Supervoxels

Stine Hansen, Srishti Gautam, Robert Jenssen, and Michael Kampffmeyer

Medical Image Analysis, 2022

Medical Image Analysis 78 (2022) 102385

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Anomaly detection-inspired few-shot medical image segmentation through self-supervision with supervoxels☆

Stine Hansen*, Srishti Gautam, Robert Jenssen, Michael Kampffmeyer

Department of Physics and Technology, UiT The Arctic University of Norway, Tromsø NO-9037, Norway

ARTICLE INFO

Article history:
Received 22 June 2021
Revised 20 January 2022
Accepted 1 February 2022
Available online 11 February 2022

Keywords:
Organ segmentation
Cardiac segmentation
Few-shot learning
Anomaly detection
Self-supervision
Supervoxels

ABSTRACT

Recent work has shown that label-efficient few-shot learning through self-supervision can achieve promising medical image segmentation results. However, few-shot segmentation models typically rely on prototype representations of the semantic classes, resulting in a loss of local information that can degrade performance. This is particularly problematic for the typically large and highly heterogeneous background class in medical image segmentation problems. Previous works have attempted to address this issue by learning additional prototypes for each class, but since the prototypes are based on a limited number of slices, we argue that this ad-hoc solution is insufficient to capture the background properties. Motivated by this, and the observation that the foreground class (e.g., one organ) is relatively homogeneous, we propose a novel anomaly detection-inspired approach to few-shot medical image segmentation in which we refrain from modeling the background explicitly. Instead, we rely solely on a single foreground prototype to compute anomaly scores for all query pixels. The segmentation is then performed by thresholding these anomaly scores using a learned threshold. Assisted by a novel self-supervision task that exploits the 3D structure of medical images through supervoxels, our proposed anomaly detection-inspired few-shot medical image segmentation model outperforms previous state-of-the-art approaches on two representative MRI datasets for the tasks of abdominal organ segmentation and cardiac segmentation.

1. Introduction

Many applications in medical image analysis, such as diagnosis (Tsochatzidis et al., 2021), treatment planning (Chen et al., 2021), and quantification of tissue volumes (Abdeltawab et al., 2020) rely heavily on semantic segmentation. To lessen the burden on the medical practitioners performing these manual, slice-by-slice segmentations, the use of deep learning for automatic segmentation has a great potential. Unfortunately, existing segmentation frameworks (Ronneberger et al., 2015; Li et al., 2018; Isensee et al., 2021) depend on supervised training and large amounts of densely labeled data, which are often unavailable in the medical domain. Moreover, their generalization properties to previously unseen classes are typically poor, necessitating the collection and labeling of new data to re-train for new tasks. Due to the huge number of potential segmentation tasks in medical images, this makes these models impractical to use.

Inspired by how humans learn from only a handful of instances, few-shot learning has emerged as a learning paradigm to foster models that can easily adapt to new concepts when exposed to just a few new, labeled samples. These models typically follow an episodic framework (Vinyals et al., 2016) where, in each episode, k labeled samples, called the support set, are used to segment the unlabeled query image(s). The models are trained on one set of classes and learn to, with only a few annotated examples, segment objects from new classes. A *trained* few-shot segmentation (FSS) model is thus able to segment an unseen organ class based on just a few labeled instances. However, in order to avoid over-fitting, typical FSS models rely on training data containing a large set of labeled training classes, generally not available in the medical domain.

In a recent work, Ouyang et al. (2020) proposed a label-efficient approach to medical image segmentation, building on metric-learning based prototypical FSS (Liu et al., 2020b; Wang et al., 2019). They suggest a model that follows the traditional

☆ All the authors are with the UiT Machine Learning Group (machine-learning.uit.no) and with *Visual Intelligence*, a Norwegian Centre for Research-based Innovation (visual-intelligence.no).

* Corresponding author.

E-mail addresses: s.hansen@uit.no (S. Hansen), srishti.gautam@uit.no (S. Gautam), robert.jenssen@uit.no (R. Jenssen), michael.c.kampffmeyer@uit.no (M. Kampffmeyer).

https://doi.org/10.1016/j.media.2022.102385

few-shot episodic framework, where class-wise prototypes are extracted from the labeled support set and used to reduce the segmentation of the unlabeled query image to a pixel-wise prototype matching in the embedding space. Whereas traditional few-shot learning models require a set of annotated training classes, Ouyang et al. (2020) propose a clever way to bypass this need by employing self-supervised training (Jing and Tian, 2020). Instead of sampling labeled support and query images, they construct the training episodes based on *one* unlabeled image slice and its corresponding superpixel (Ren and Malik, 2003) segmentation: One randomly sampled superpixel serves as foreground mask, and together with the original image slice, these form the support image-label pair. The query pair is then constructed by applying random transformations to the support pair. In this way, they enable training of the network without using annotations, i.e. the model is trained unsupervised. Finally, in the inference phase, they only need a few labeled image slices to perform segmentation on new classes.

However, a general problem with prototypical FSS is the loss of local information caused by average pooling of features during prototype extraction. This is particularly problematic for spatially heterogeneous classes like the background class in medical image segmentation problems, which can contain any semantic class other than the foreground class. Previous metric-learning based works have addressed this issue by computing additional prototypes per class to capture more diverse features. Liu et al. (2020b) clustered the features within each class to obtain *part-aware* prototypes and in the current state-of-the-art method, Ouyang et al. (2020) computed additional *local* prototypes on a regular grid.

We argue that it is insufficient to model the entire background volume with prototypes estimated from a few support slices and propose a conceptually different approach where we do *not* increase the number of background prototypes but remove the need for these altogether. Inspired by the anomaly detection literature (Chandola et al., 2009; Ruff et al., 2021), we propose to only model the relatively homogeneous foreground class with a single prototype and introduce an anomaly score that measures the dissimilarity between this foreground prototype and all query pixels. Segmentation is then performed by thresholding the anomaly scores using a learned threshold that encourages compact foreground representations. For direct comparison of our novel anomaly detection-inspired few-shot medical image segmentation method to that of Ouyang et al. (2020) and other representative works, our baseline setup follows their approach, working with 2D image slices. Within the existing 2D setup, we, as an added contribution, propose a new self-supervision task by extending the superpixel-based self-supervision scheme by Ouyang et al. (2020) to 3D in order to utilize the volumetric nature of the data. As a natural extension, facilitated by the new self-supervision task, we further indicate potential benefits beyond this 2D setup by exploring a direct 3D treatment of the problem by employing a 3D convolutional neural network (CNN) as embedding network.

By only explicitly modeling the foreground class, we argue that our proposed approach is more robust to background outside the support slices, compared to current state-of-the-art methods (Ouyang et al., 2020; Roy et al., 2020). To further illustrate this, we introduce a new evaluation protocol where we, based on labeled slices from the support image, segment the entire query image, thus being more exposed to background effects. Previous works, on the other hand, limit the evaluation of the query image only to the slices containing the class of interest. However, this approach requires additional weak labels in the form of information about the location of the class in the query image, which is unrealistic and cumbersome, especially in the medical setting.

In summary, the main contributions of this work are three-fold. We propose:

(1) A simple but effective anomaly detection-inspired approach to FSS that outperforms prior state-of-the-art methods and removes the need to learn a large number of prototypes.
(2) A novel self-supervision task that exploits the 3D structural information in medical images within the 2D setup and indicate the potential of training 3D CNNs for direct volume segmentation.
(3) A new evaluation protocol for few-shot medical image segmentation that does not rely on weak-labels and therefore is more applicable in practical scenarios.

2. Related work

2.1. Few-shot meta-learning

As opposed to classical supervised learning that specializes a model to perform one specific task by optimizing over training samples, few-shot meta-learning optimizes over a set of training tasks, with the goal of obtaining a model that can quickly adapt to new, unseen tasks. There exist various approaches to few-shot learning, including i) learning to fine-tune (Finn et al., 2017; Ravi and Larochelle, 2017), ii) sequence based (Mishra et al., 2018; Santoro et al., 2016), and iii) metric-learning based approaches (Vinyals et al., 2016; Snell et al., 2017; Nguyen et al., 2020). Due to its simplicity and efficiency, the latter category has recently received a lot of attention, and the models relevant for this paper build on this principle. Vinyals et al. (2016) combined deep feature learning with non-parametric methods in the Matching Network, by performing weighted nearest-neighbor classification in the embedding space. They proposed to train the model in episodes where a small labeled support set and an unlabeled query image are mapped to the query label, making the model able to adapt to unseen classes without the need for fine-tuning. Whereas the Matching Network only performed one-shot image classification, Snell et al. (2017) later proposed the Prototypical Network, which extended the problem to include few-shot classification. Based on the idea that there exists an embedding space, in which samples cluster around their class prototype representation, they proposed a simpler model with a shared encoder between the support and query set, and a nearest-neighbor prototype matching in the embedding space.

2.2. Few-shot semantic segmentation

Few-shot semantic segmentation extends few-shot image classification (Vinyals et al., 2016; Snell et al., 2017; Nguyen et al., 2020) to pixel-level classifications (Shaban et al., 2017; Rakelly et al., 2018; Zhang et al., 2020; Wang et al., 2019), and the goal is to, based on a few densely labeled samples from one (or more) new class(es), segment the class(es) in a new image. A recent line of work builds on the ideas from the Prototypical Network by Snell et al. (2017), and can be roughly split into two groups: models where predictions are based directly on the cosine similarity between query features and prototypes in the embedding space (Wang et al., 2019; Liu et al., 2020b; Ouyang et al., 2020), and models that find the correlation between query features and prototypes by employing decoding networks to get the final prediction (Dong and Xing, 2018; Zhang et al., 2019; Liu et al., 2020a; Li et al., 2021a; Zhang et al., 2021; Tian et al., 2020).

Dong and Xing (2018) first adopted the idea of metric-learning based prototypical networks to perform few-shot semantic segmentation. They proposed a two-branched model: a prototype learner, learning class-wise prototypes from the labeled support

set, and a segmentation network where the prototypes were used to guide the segmentation of the query image. Most relevant for this work, Wang et al. (2019) argued that parametric segmentation generalizes poorly, and proposed the Prototype Alignment Network (PANet), a simpler model where the knowledge extraction and segmentation process is separated. By exploiting prototypes extracted from the semantic classes of the support set, they reduced the segmentation of the query image to a non-parametric pixel-wise nearest-neighbor prototype matching, thereby creating a new branch of FSS models. Building on PANet, (Liu et al., 2020b) addressed the limitation of reducing semantic classes to a simple prototype and proposed the Part-aware Prototype Network (PPNet), where each semantic class is represented by multiple prototypes to capture more diverse features. Liu et al. (2020b) further adopted a semantic branch for parametric segmentation during training to learn better representations. Ouyang et al. (2020) adapted ideas from PANet to perform FSS in the medical domain. They addressed the major restricting factor preventing medical FSS, e.g the dependency on a large a set of annotated training classes. This barrier was overcome by the introduction of a superpixel-based self-supervised learning scheme, enabling the training of FSS networks without the need for labeled data. Ouyang et al. (2020) further introduced the Adaptive Local Prototype pooling enpowered prototypical Network (ALPNet) where additional *local* prototypes are computed on a regular grid to preserve local information and enhance segmentation performance.

A different approach to medical FSS was suggested by Roy et al. (2020), and was the first FSS model for medical image segmentation. Their proposed SE-Net employs squeeze and excite blocks (Hu et al., 2018) in a two-armed architecture consisting of one conditioner arm, processing the support set, and one segmenter arm, interacting with the conditioner arm to segment the query image. However, this model is trained supervised, requiring a set of labeled classes for training.

Based on our experience, training a decoder in a self-supervised setting, where the training task (superpixel segmentation) differs from the inference task (organ segmentation), is challenging and leads to performance degradation. In this paper, we thus, partially inspired by the state-of-the-art model (Ouyang et al., 2020), build further on the branch initiated by Wang et al. (2019) to perform FSS in the medical domain. We propose a novel FSS model that, unlike previous approaches in this branch (Wang et al., 2019; Liu et al., 2020b; Ouyang et al., 2020), does *not* explicitly model the complex background class, but relies solely on one foreground prototype.

2.3. Self-supervised learning

When large labeled datasets are not available, self-supervision can be used to learn representations by training the deep learning model on an auxiliary task that is defined such that the label is *implicitly* available from the data. A good auxiliary task should require high-level image understanding to be solved, thereby encouraging the network to encode this type of information. Commonly used auxiliary tasks include image inpaining (Larsson et al., 2016; Pathak et al., 2016; Zhang et al., 2016), contrastive learning (Chen et al., 2020; Misra and Maaten, 2020), rotation prediction (Komodakis and Gidaris, 2018), solving jigsaw puzzles (Noroozi and Favaro, 2016), and relative patch location prediction (Doersch et al., 2015).

In the medical domain, self-supervised learning (SSL) has been used to improve performance on other (main) tasks by exploiting unlabeled data in a multi-task learning setting (Chen et al., 2019; Li et al., 2021b) and to pre-train models before transferring them to new (main) tasks (Bai et al., 2019; Zhu et al., 2020; Dong et al., 2021; Lu et al., 2021). In Ouyang et al. (2020), SSL was used to train a FSS model completely unsupervised using a novel superpixel-based auxiliary task, removing the need for labeled data during training. We build on this work by extending the proposed self-supervision scheme to 3D supervoxels.

2.4. Supervoxel segmentation

Supervoxels and superpixels are groupings of local voxels/pixels in an image that share similar characteristics. The boundaries of a supervoxel/superpixel therefore tend to follow the boundaries of the structures in the image, providing natural sub-regions. Supervoxel and superpixel segmentation has become a common tool in computer vision, also in the medical domain (Huang et al., 2020; Irving et al., 2016). For a detailed comparison of available superpixel segmentation algorithms, we refer the reader to (Stutz et al., 2018).

3. Problem definition

Given a labeled dataset with classes $\mathcal{C}_{train}$ (here: $\mathcal{C}_{train} = \{supervoxel_1, supervoxel_2, \ldots\}$), FSS models aim to learn a quick adaption to new classes $\mathcal{C}_{test}$ (e.g. $\mathcal{C}_{test} = \{liver, kidney, spleen\}$) when exposed to only a few labeled samples. The training and testing are performed in an episodic manner (Vinyals et al., 2016) where, in each episode, N classes are sampled from $\mathcal{C}$ to create a support set and a query set. The input to an episode is the support image(s) (with annotations) and a query image, and the output is the predicted query mask. In an N-way k-shot setting, the support set $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_{N\times k}, \mathbf{y}_{N\times k}))\}$ consists of k image slices $\mathbf{x} \in \mathbb{R}^{H\times W}$ (with annotations $\mathbf{y} \in \mathbb{R}^{H\times W}$ indicating the class of each pixel) from each of the N classes, whereas the query set consists of one query image $\mathcal{Q} = \left\{(\mathbf{x}_1^*, \mathbf{y}_1^*)\right\}$ containing one or more of the N classes.

4. Methods

In this work, we propose an anomaly detection-inspired network (ADNet) for prototypical FSS[1]. We employ a shared feature extractor between the support and query images and perform metric learning-based segmentation in the embedding space. Unlike prior approaches that obtain prototypes for both foreground *and* background classes (Liu et al., 2020b; Ouyang et al., 2020; Wang et al., 2019), we only consider foreground prototypes to avoid the aforementioned problems related to explicitly modeling the large and heterogeneous background class. Based on *one* foreground prototype, we compute anomaly scores for all query feature vectors. The segmentation of the query image is then based on these anomaly scores and a learned anomaly threshold. To train our model, we take inspiration from Ouyang et al. (2020) and propose a new supervoxel-based self-supervision pipeline. Fig. 1 and Fig. 2 provide an overview of the model during training and inference, respectively.

4.1. Anomaly detection-inspired few-shot segmentation

We denote the encoding network as f_θ and start by embedding the support and query images into deep features, $f_\theta(\mathbf{x}) = F^s$ and $f_\theta(\mathbf{x}^*) = F^q$, respectively. As opposed to previous works, we are only interested in explicitly modeling the foreground in each episode. We do this by employing the segmentation mask to perform masked average pooling (MAP), but only for the foreground class c. We resize the support feature map F^s to the mask size

[1] By "anomaly" we refer to abnormalities compared to our defined normal class (foreground), and not necessarily something that occurs infrequently.

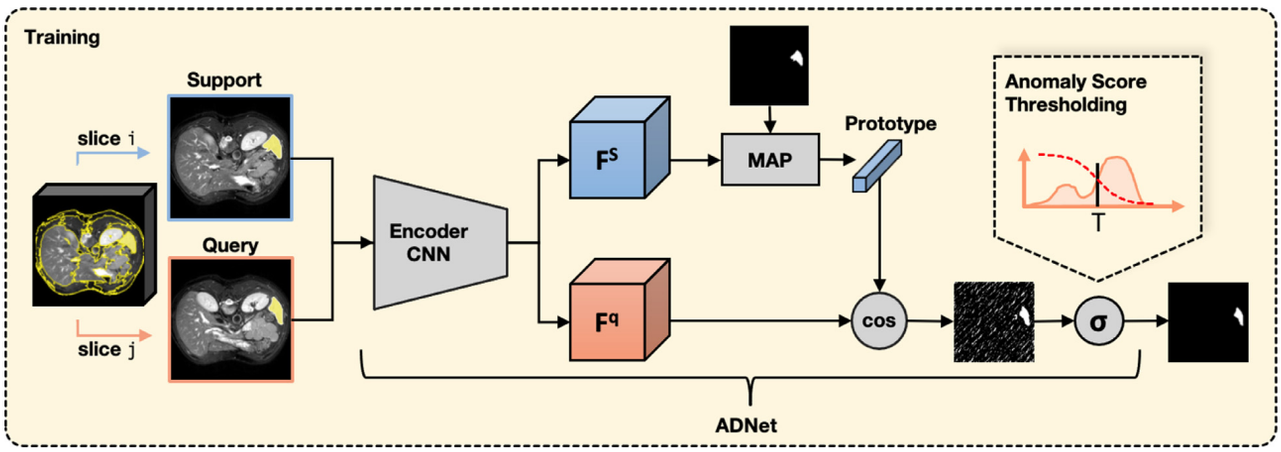


Fig. 1. Illustration of the model during training. Support and query slices are obtained from the same image volume as two different 2D slices containing a randomly sampled supervoxel. A shared feature encoder encodes the query and the support images into deep feature maps. The support features are then resized to the mask size and masked average pooling is applied to compute the foreground prototype. For each query feature vector, an anomaly score is computed based on the cosine similarity to the prototype. Finally, the segmentation of the query image is performed by thresholding the anomaly scores using a learned anomaly threshold.

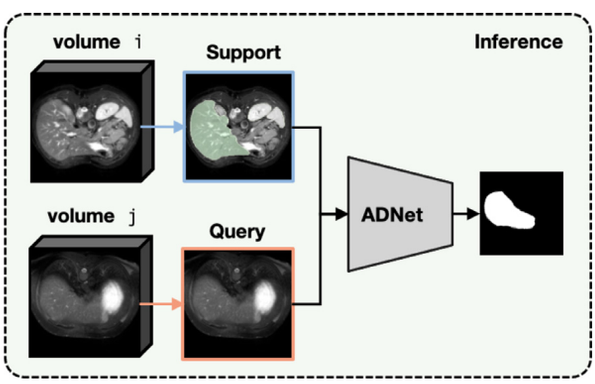


Fig. 2. Illustration of the model during inference. Based on labeled slices from the support volume, the query volume is segmented slice by slice, one class at a time.

(H, W) and compute one foreground prototype $p \in \mathbb{R}^d$, where d is the dimension of the embedding space:

$$p = \frac{\sum_{x,y} F^s(x,y) \odot \mathbf{y}^{fg}(x,y)}{\sum_{x,y} \mathbf{y}^{fg}(x,y)}, \tag{1}$$

where $\odot$ denotes the Hadamard product and $\mathbf{y}^{fg} = \mathbb{1}(\mathbf{y} = c)$ is the binary foreground mask of class c[2].

To segment the query image based on this *one* class-prototype, we design a threshold-based metric learning approach to the segmentation. We first obtain an anomaly score S for each query feature vector $F^q(x,y)$ by calculating the (negative) cosine similarity to the foreground prototype p of the episode:

$$S(x,y) = -\alpha \frac{F^q(x,y) \cdot p}{\|F^q(x,y)\| \|p\|}, \tag{2}$$

where $\alpha = 20$ is a scaling factor introduced by Oreshkin et al. (2018). In this way, query feature vectors that are identical to the prototype will get an anomaly score of $-\alpha$ (minimum), whereas query feature vectors that are pointing in the opposite direction, relative to the prototype, get an anomaly score of α (maximum). The predicted foreground mask is then found by thresholding these anomaly scores with a learned parameter T. To make the process differentiable, we perform soft thresholding by applying a shifted Sigmoid:

$$\hat{\mathbf{y}}^q_{fg}(x,y) = 1 - \sigma(S(x,y) - T), \tag{3}$$

where $\sigma(\cdot)$ denotes the Sigmoid function with a steepness parameter $\kappa = 0.5$. The impact of the steepness parameter is examined in Section 5.3.4. In this way, query feature vectors with an anomaly score below T (similar to the prototype) get a foreground probability above 0.5, whereas query feature vectors with an anomaly score above T (dissimilar to the prototype) get a foreground probability below 0.5. The predicted background mask is finally found as $\hat{\mathbf{y}}^q_{bg} = 1 - \hat{\mathbf{y}}^q_{fg}$.

The predicted foreground and background masks for the query image are then upsampled to the image size (H, W) and we compute the binary cross-entropy segmentation loss:

$$\mathcal{L}_S = -\frac{1}{HW} \sum_{x,y} \mathbf{y}^q_{bg}(x,y) \log(\hat{\mathbf{y}}^q_{bg}(x,y)) + \mathbf{y}^q_{fg}(x,y) \log(\hat{\mathbf{y}}^q_{fg}(x,y)). \tag{4}$$

In order to encourage a compact embedding of the foreground classes, we construct an additional loss term $\mathcal{L}_T = T/\alpha$ that minimizes the learned threshold. The effect of this loss component is examined in Section 5.3.2.

Following common practice (Liu et al., 2020b; Ouyang et al., 2020; Wang et al., 2019), we also add a prototype alignment regularization loss where the roles of support and query are reversed. The *predicted* query mask is used to compute a prototype that segments the support image:

$$\mathcal{L}_{PAR} = -\frac{1}{HW} \sum_{x,y} \mathbf{y}^s_{bg}(x,y) \log(\hat{\mathbf{y}}^s_{bg}(x,y)) + \mathbf{y}^s_{fg}(x,y) \log(\hat{\mathbf{y}}^s_{fg}(x,y)). \tag{5}$$

This gives us the overall loss function

$$\mathcal{L} = \mathcal{L}_S + \mathcal{L}_T + \mathcal{L}_{PAR}. \tag{6}$$

4.2. Supervoxel-based self-supervision

The ADNet is parameterized by $\mathcal{P} = \{\theta, T\}$ and trained self-supervised (unsupervised) end-to-end in an episodic manner. For ease of comparison to previous approaches, our baseline setup follows a 2D approach, where volumes are segmented slice-by-slice. However, to better utilize the volumetric nature of the medical images, we propose a new self-supervision task that exploits 3D supervoxels during the model's training phase. As supervoxels are

[2] $\mathbb{1}(\cdot)$ is the indicator function, returning 1 if the argument is true and 0 otherwise.

sub-volumes of the image, representing groups of similar voxels in local regions of the image volume, this allows us to sample 3D pseudo-segmentation masks for semantically uniform regions in the image.

In the training phase, each episode is constructed based on one unlabeled image volume and its supervoxel segmentation: First, one random supervoxel is sampled to represent the foreground class, resulting in a binary 3D segmentation mask. Then, we sample two 2D slices from the image containing this "class"/supervoxel to serve as support and query images. By exploiting the relations across slices, we are able to increase the amount of information that can be extracted in the self-supervision task compared to prior approaches. Following Ouyang et al. (2020), we additionally apply random transformations to one of the images (query or support) to encourage invariance to shape and intensity differences.

The supervoxels for all image volumes are computed offline using a 3D extension of the same unsupervised segmentation algorithm (Felzenszwalb and Huttenlocher, 2004) as in (Ouyang et al., 2020). This is an efficient graph-based image segmentation algorithm building on euclidean distances between neighboring pixels. In the 3D extension, this corresponds to the distances from each voxel to its 26 nearest neighbours. In medical images, the resolution in z-direction (slice thickness) is typically different from the in-plane (x, y) resolution. To account for this anisotropic voxel resolution, we re-weight all distances along the z-direction ($xz-$, $yz-$ and $xyz-$direction) according to the spatial ratios.

The supervoxel generation has one hyper-parameter ρ that controls the minimum supervoxel size, where a larger ρ corresponds to larger and fewer supervoxels. The effect of this parameter on the final segmentation result is examined in Section 5.3.3.

4.3. Implementation details

The implementation is based on the PyTorch (v1.7.1) implementation of SSL-ALPNet (Ouyang et al., 2020). The encoder network used in all the 2D experiments is a ResNet-101 pretrained on MS-COCO, where the classifier is replaced by a 1×1 convolutional layer to reduce the feature dimension from 2048 to 256. Following ALPNet, we optimize the loss using stochastic gradient descent with momentum 0.9, a learning rate of 1e-3 with a decay rate of 0.98 per 1k epochs, and a weight decay of 5e-4 over 50k iterations. To address the class imbalance, we follow previous work and weigh the foreground and background class in the cross-entropy loss (1.0 and 0.1, respectively). To further stabilize training, we set a minimum threshold of 200 pixels on the supervoxel size in the slices sampled as support/query. Supervoxel generation is done offline (once per image volume) and is relatively computationally efficient[3]. Training takes 1.8h on a Nivida RTX 2080Ti GPU.

5. Experiments

5.1. Setup

5.1.1. Data

We assess the proposed method on representative publicly available datasets[4]:

(1) **MS-CMRSeg** (bSSFP fold), from the MICCAI 2019 Multi-sequence Cardiac MRI Segmentation Challenge, containing 35 3D cardiac MRI scans with on average 13 slices (Zhuang, 2018; 2016).

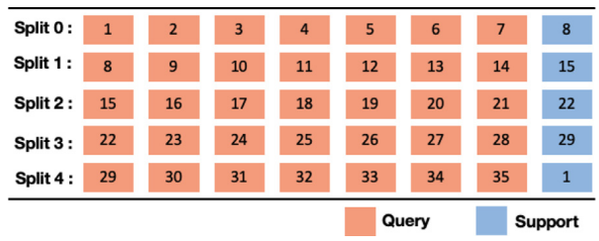


Fig. 3. Setup for the five-fold cross-validation. This illustrates how the patient IDs are distributed among the splits and how the support/query volumes are selected for the cardiac MRI dataset. For each fold, a model is trained on all images *not* present in that fold. During inference, the left-out fold is used exclusively, where the labeled support image is exploited to segment the query images slice by slice, class by class. The CHAOS dataset is split into five folds in a similar manner.

(2) **CHAOS**, from the ISBI 2019 Combined Healthy Abdominal Organ Segmentation Challenge (task 5), containing 20 3D T2-SPIR MRI scans with on average 36 slices (Kavur et al., 2021; 2019; 2020).

To compare our results to Ouyang et al. (2020), we follow the same pre-processing scheme: 1) Cut the top 0.5% intensities. 2) Resample image slices (short-axis slices for the cardiac images and axial slices for the abdominal images) to the same spatial resolution. 3) Crop slices to unify size (256×256 pixels). Further, to fit into the pretrained network, each slice is repeated three times along the channel dimension.

In all experiments, the models are trained self-supervised (unsupervised) and evaluated in a five-fold cross-validation manner, where, in each fold, the support images are sampled from *one* of the patients and the remaining patients are treated as query (see Fig. 3). Furthermore, to account for the stochasticity in the model and optimization, we repeat each fold three times. In the cardiac MRI scans we segment three classes: Left-ventricle blood pool (LV-BP), left-ventricle myocardium (LV-MYO) and right-ventricle (RV). In the abdominal MRI scans, we segment four classes: left kidney (L. kid.), right kidney (R. kid.), liver, and spleen. Following previous methods (Ouyang et al., 2020; Roy et al., 2020), each class is segmented separately in binary foreground/background segmentation problems[5]. Since the models are trained self-supervised, we do *not* exclude image slices that contain the target classes.

5.1.2. Evaluation metric

Following common practice (Ouyang et al., 2020; Roy et al., 2020) we employ the mean dice score to compare the model predictions to the ground truth segmentations. The dice score, D, between two segmentations A and B is given by

$$D(A, B) = 2\frac{|A \cap B|}{|A| + |B|} \cdot 100\%, \tag{7}$$

meaning that a dice score of 100% corresponds to a perfect match between the segmentations.

5.1.3. Evaluation protocols

During inference, the query volumes are segmented episode-wise, slice-by-slice, based on labeled support slices. For this reason, it is necessary to define an evaluation protocol that describes how to construct the episodes during inference, i.e. how to pair support and query images in episodes. In the experiments, we evaluate all models under two different evaluation protocols (EPs), illustrated in Fig. 4.

[3] The compute time for generating all supervoxels for the MS-CMRSeg dataset is less than 3 minutes using a Quad-Core Intel Core i7 processor.

[4] Links to public datasets: MS-CMRSeg and CHAOS

[5] As the segmentation only relies on the computation of the cosine similarity to a class-specific prototype and a threshold which is shared among classes, the proposed method may be extended to account for multi-class scenarios. A detailed analysis of this is left for future work.

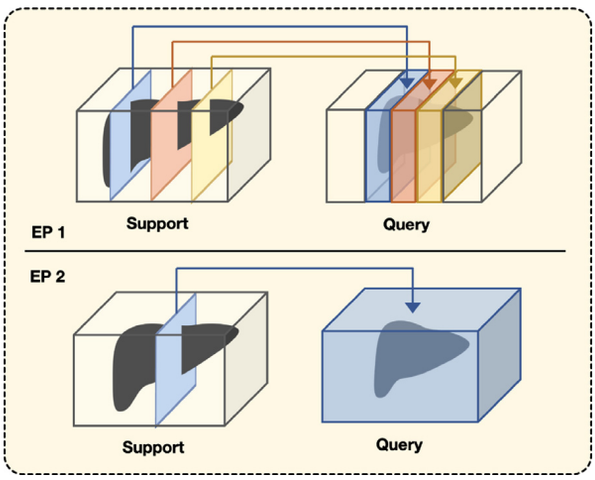


Fig. 4. Illustration of EP1 (top) and EP2 (bottom). In EP1, the support and query volumes are divided into three succeeding sub-chunks. The middle slice in each sub-chunk of the support volume is labeled and used to segment all the slices in the corresponding sub-chunk in the query volume. This means that the protocol requires weak labels indicating where the class of interest is located in the query volume. In EP2, the middle slice of the support volume is labeled and used to segment *all* slices in the query volume, avoiding the need for additional weak labels.

Evaluation protocol 1 (EP1) Previous works (Ouyang et al., 2020; Roy et al., 2020) follow an evaluation protocol that requires weak labels for all query images, i.e. there is a need to indicate (label) in which slices the foreground class is located. For a given class to be segmented, the chunk of slices in both the support and query volumes containing this class is divided into three succeeding sub-chunks. The middle slice in each sub-chunk of the support volume is used to segment all the slices in the corresponding sub-chunk in the query. In practice, this requires manual and time-consuming input from medical experts during the inference phase, where they have to scroll through each query image volume to mark the slices containing the class(es) of interest.

Evaluation protocol 2 (EP2) To avoid the need for weak query labels during inference, we introduce a new evaluation protocol that does not depend on the position of the target volume, and thus is more applicable in practical situations. Here, we simply sample $k = 1$ slices from the support foreground volume and use this information to segment the entire query volume. To limit boundary effects, we choose the middle slice of the support foreground volume.

5.2. Comparison to state-of-the-art

We compare our model to three modern FSS models: PANet (Wang et al., 2019), ALPNet Ouyang et al. (2020), and PPNet (Liu et al., 2020b) with five (default) prototypes per class. Additionally, to compare our one-prototype anomaly approach to a one-prototype decoder approach, we adopt the dense comparison module proposed in (Zhang et al., 2019) as a decoder on top of the backbone network and refer to this network as CANet[6].

The current state-of-the-art method for medical FSS, Ouyang et al. (2020), showed that training PANet and ALPNet in a self-supervised manner improved the dice scores of the segmentation results considerably, compared to classical supervised FSS. Specifically, the dice scores on the MS-CMRSeg and CHAOS datasets increased by an average of 17.9 and 26.1 percentage points, respectively. Here, we are thus only focusing on SSL approaches. pSSL refers to the superpixel SSL approach presented in Ouyang et al. (2020), whereas vSSL refers to our proposed supervoxel-based approach.

Table 1 and Table 2 present the results under EP1 and EP2, respectively, as mean and standard deviations over three runs (over all splits). Summarized details about the models can be found in Table 3.

In Table 1 we can see that our proposed model under EP1 performs similarly to the state-of-the-art on both datasets, while using significantly fewer prototypes compared to the closest competitors. We can also observe that the models that use just a few prototypes to model the background (PANet, PPNet) perform poorly and are among the three worst performing models for both datasets. Furthermore, by only modeling the foreground class and segmenting the query image using a decoding network, CANet results in the lowest (overall) dice score on the cardiac dataset.

In a more realistic scenario, information about the location of the foreground volume in the query images is typically not available. We therefore evaluate the models under EP2 (Table 2) and we observe that our proposed approach outperforms the state-of-the-art. One-sided Wilcoxon signed rank tests (Wilcoxon, 1992) on the mean dice scores across all runs indicate a significant difference between the segmentation results obtained from vSSL-ADNet and pSSL-ALPNet for both datasets under EP2 ($p < 0.05$). For the abdominal data, our model improves the segmentation results by more than 20 percentage points compared to pSSL-ALPNet. The main reason for this large improvement is that we now have to consider *all* the query slices (not only the slices containing the organ to be segmented), meaning that the background class is much larger and much more diverse. This again complicates the task of modeling the background with prototypes, whereas our anomaly detection-inspired model without background prototypes is less affected. The somewhat lower performance and high standard deviation for left-kidney and spleen are related to the weak boundaries between these organs (see discussion in Section 6). Furthermore, we obtain considerable, but smaller, improvements on the cardiac dataset under EP2. This is related to the lower number of slices and the less diverse background in these images, making the task of modeling the background with prototypes less complicated. Qualitative comparisons are provided in Fig. 5 and Fig. 6, where we can see that our approach is less prone to over-segmentation.

5.3. Model analysis

5.3.1. Analysis of learned threshold

To evaluate the learned threshold's precision on the unseen test data, we have conducted a line search where we, in the inference phase, evaluate the dice score obtained using a range of different thresholds between -20 and -15. The experiment was performed on three runs for each split and the mean dice score and standard deviation (shaded region) are reported in Fig. 7. The learned threshold is averaged over all runs and and represented by the vertical black line[7]. From the plot, we see that the threshold optimized for the training data is close to the ideal threshold for the test data, with little to gain in terms of increased dice score.

5.3.2. Ablation study

To evaluate the effect of the three components of our loss function, we conduct an ablation study on the cardiac dataset. Table 4 illustrates that $\mathcal{L}_T$ and $\mathcal{L}_{PAR}$ improve the dice score across all classes. Further, Fig. 8 shows qualitatively the effect of $\mathcal{L}_T$ on the segmentation of one image slice from the MS-CMRSeg dataset. Here, it can be seen how the encouraging of a more compact foreground embedding via $\mathcal{L}_T$ reduces the over-segmentation, especially for the left-ventricle myocardium.

[6] Code available: PANet, ALPNet, PPNet, and CANet.

[7] The small, gray shaded region indicates the range of learned threshold values.

Table 1
Mean dice score and standard deviation over three runs per split under EP1.

Method	Cardiac MRI				Abdominal MRI				
	LV-BP	LV-MYO	RV	Mean	L kid.	R kid.	Spleen	Liver	Mean
pSSL-PANet	80.20±4.39	45.67±2.58	66.95±4.65	64.27±14.23	63.09±9.31	66.09±8.73	63.93±8.65	72.08±3.83	66.30±3.51
pSSL-ALPNet	**87.54 ± 1.63**	60.19±4.55	76.08±4.72	74.60±11.21	**81.00 ± 4.01**	**84.66 ± 2.40**	72.32±7.69	75.89±3.02	78.46±4.72
vSSL-PPNet	67.78±8.31	42.61±6.16	60.80±6.44	57.06±10.61	62.13±7.85	71.78±11.04	66.57±9.04	73.12±2.51	68.40±4.37
vSSL-CANet	78.99±4.72	43.61±3.38	61.10±3.60	61.07±14.64	69.53 ± 12.05	77.15 ± 10.71	67.05 ± 6.87	72.88 ± 3.27	71.65 ± 3.79
vSSL-ADNet	87.53±2.03	**62.43 ± 3.98**	**77.31 ± 3.48**	**75.76 ± 10.31**	75.28±14.80	83.28±13.36	**75.92 ± 8.90**	**80.81 ± 2.36**	**78.82 ± 3.35**

Table 2
Mean dice score and standard deviation over three runs per split under EP2. * indicates that the increase in mean dice score for the best performing model is statistically significant ($p < 0.05$).

Method	Cardiac MRI				Abdominal MRI				
	LV-BP	LV-MYO	RV	Mean	L kid.	R kid.	Spleen	Liver	Mean
pSSL-PANet	68.28±5.67	38.60±3.72	55.22±5.18	54.03±12.15	32.85±6.74	30.18±4.85	34.82±8.52	53.89±3.15	37.94±9.36
pSSL-ALPNet	80.65±3.93	53.31±6.31	**69.25 ± 2.80**	67.74±11.21	56.42±5.74	50.37±7.77	44.70±7.77	56.73±3.07	52.05±4.94
vSSL-PPNet	56.69±8.35	34.78±7.30	47.60±6.07	46.35±8.99	43.36±10.44	56.94±14.39	43.06±9.73	56.32±7.57	49.92±6.71
vSSL-CANet	74.54±4.20	35.08±4.30	47.65±5.73	52.42±16.46	50.18 ± 13.02	69.91 ± 12.84	48.84 ± 8.61	64.00 ± 3.44	58.23 ± 8.98
vSSL-ADNet	**82.81 ± 3.20**	**59.46 ± 2.97**	66.58±4.74	**69.62 ± 9.77***	**62.33 ± 9.70**	**86.46 ± 2.74**	**63.73 ± 11.66**	**77.12 ± 3.41**	**72.41 ± 9.96***

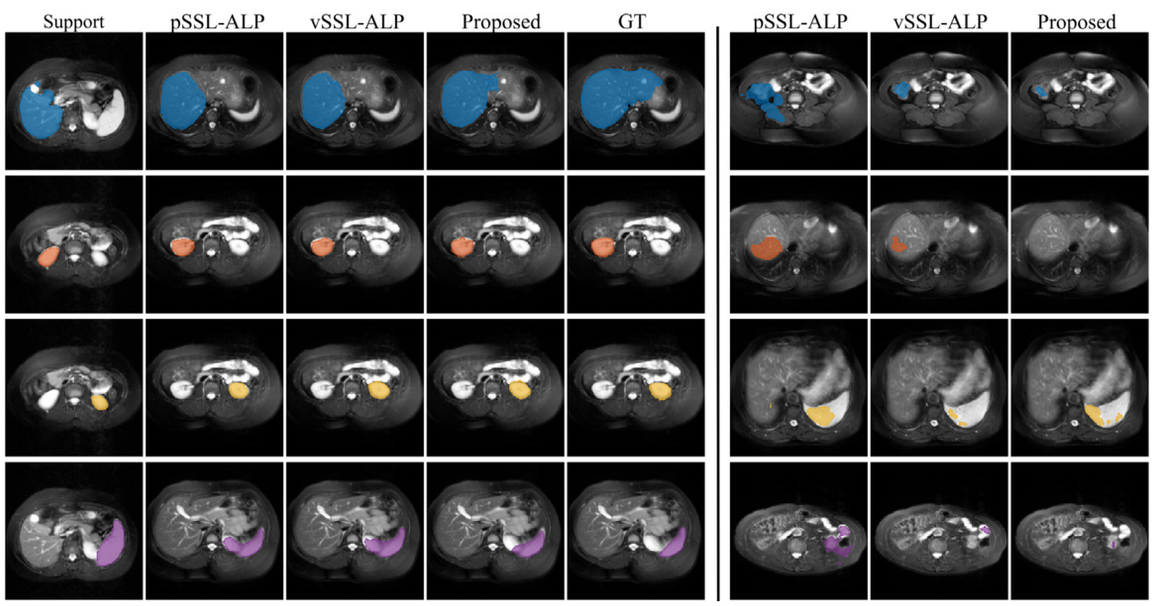


Fig. 5. Qualitative comparisons for the abdominal MRI dataset. To the left of the solid line, we see (left to right) the support image, the segmentation results of a query slice containing the foreground class, and the ground truth segmentation of this query image. To the right, we see segmentation results for query slices *not* containing the foreground class. Top to bottom: liver, right kidney, left kidney, and spleen. The proposed method is more robust to background outside the support slice, resulting in less over-segmentation.

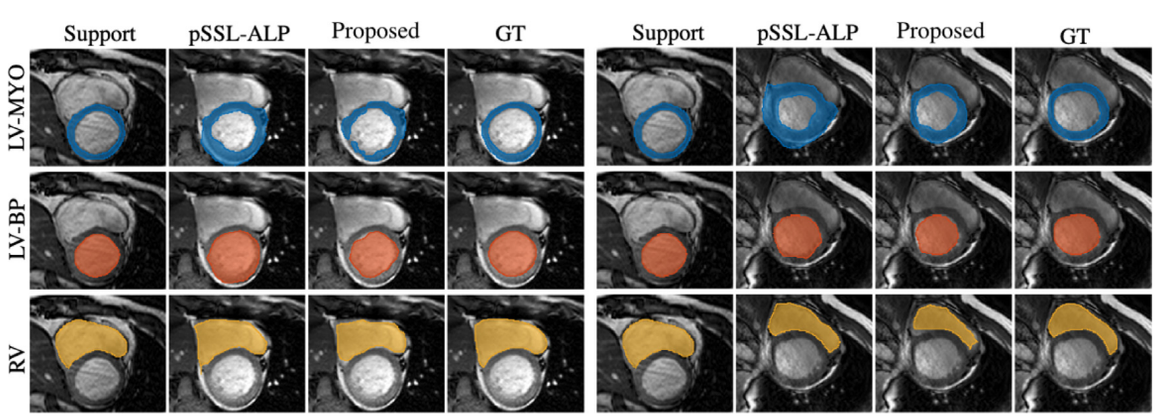


Fig. 6. Qualitative comparisons for two episodes with the same support volume from the cardiac MRI dataset. Left to right: Support image, segmentation results of a query slice, and ground truth segmentation of this query image. The segmentation results are quite similar but the proposed method captures the left-ventricle myocardium and left ventricle blood pool better, with less over-segmentation.

Table 3
Summarized information about the models. *The number of prototypes in ALPNet is adaptive and we report the average number over all classes during inference.

Method	Backbone	Decoder	# Foreground prototypes	# Background prototypes
pSSL-PANet		✗	1	1
pSSL-ALPNet*		✗	4	246
vSSL-PPNet	ResNet-101	✗	5	5
vSSL-CANet		✓	1	0
vSSL-ADNet		✗	1	0

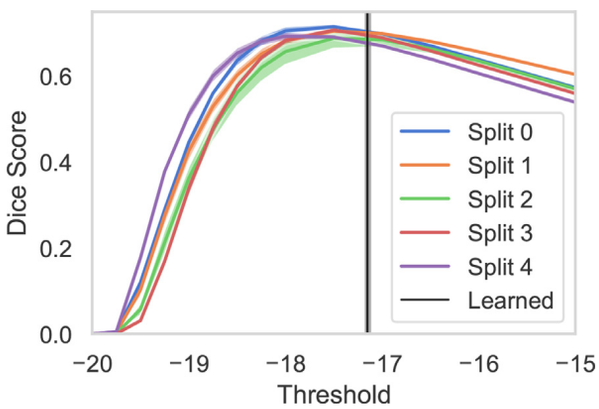


Fig. 7. Analysis of the precision of the learned threshold. The plot shows the mean dice score (with standard deviation) obtained for a range of thresholds during inference on the MS-CMRSeg dataset. The learned threshold is indicated by the black vertical line.

Table 4
Ablation study showing how the loss function components affect the results under EP1. * indicates that the increase in mean dice score for the best performing model is statistically significant ($p < 0.05$).

$\mathcal{L}_S$	$\mathcal{L}_T$	$\mathcal{L}_{PAR}$	Cardiac MRI			
			LV-BP	LV-MYO	RV	Mean
✓	✓	✓	**87.53 ± 2.03**	**62.43 ± 3.98**	**77.31 ± 3.48**	**75.76 ± 10.31***
✓		✓	87.41±2.08	58.48±3.17	74.95±3.33	73.61±11.85
✓	✓		82.80±3.26	57.70±3.05	72.63±2.43	71.05±10.31
✓			83.62±2.51	51.38±2.52	68.36±3.02	67.77±13.17

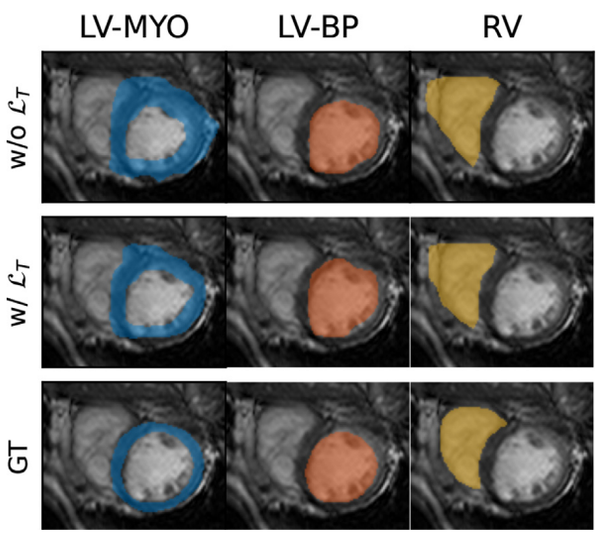


Fig. 8. Qualitative (zoomed in) segmentation results for one slice in the MS-CMRSeg dataset obtained from a model trained with (middle) and without (top) $\mathcal{L}_T$ in the total loss. The lower row shows the ground truth, and it is evident that the threshold loss reduces the over-segmentation, especially for the left-ventricle myocardium.

Table 5
Supervoxel parameter sensitivity. Analysis of the parameter controlling the minimum supervoxel size (n.o. voxels), on the cardiac MRI dataset under EP1.

ρ	Cardiac MRI			
	LV-BP	LV-MYO	RV	Mean
500	84.64±1.51	43.48±6.25	66.87±5.00	65.00±16.86
1000	**87.53 ± 2.03**	62.43±3.98	**77.31 ± 3.48**	**75.76 ± 10.31**
1500	86.91±2.47	**62.60 ± 3.44**	75.30±1.91	74.94±9.93
2000	87.30±1.80	61.21±3.33	73.92±2.51	74.14±10.65
5000	77.84 ±8.49	49.30 ±7.93	66.44 ±10.1	64.53 ±14.72

Table 6
Steepness parameter sensitivity. Analysis of the parameter controlling the sigmoid steepness parameter, on the cardiac MRI dataset under EP1.

κ	Cardiac MRI			
	LV-BP	LV-MYO	RV	Mean
0.1	80.69±4.04	17.10±5.11	52.69±10.35	50.16±26.02
0.3	**87.95 ± 1.35**	56.79±6.32	**78.44 ± 2.48**	74.39±13.04
0.5	87.54±2.03	**62.44 ± 3.98**	77.32±3.48	**75.76 ± 10.31**
0.7	87.22±2.13	62.21±2.71	76.90±3.40	75.45±10.26
0.9	85.41±2.90	60.67±3.57	76.06±3.33	74.05±10.20
1.0	85.68±2.81	59.40±4.04	75.22±4.00	73.43±10.80

5.3.3. Sensitivity of supervoxel size

A sensitivity analysis of the parameter ρ, controlling the supervoxel size, is conducted on the MS-CMRSeg dataset and the results are presented in Table 5. As shown by these results, the final segmentation performance is relatively robust for a range of minimum size values from $\rho = 1000$ to $\rho = 2000$. However, if we allow the sizes to become too small ($\rho = 500$) or too large ($\rho = 5000$), we see that the performance is negatively affected. Examples of 2D slices from the 3D supervoxel segmentations for the different values of ρ are shown in Fig. 9.

According to the sensitivity study, a reasonable value is $\rho = 1000$, and all the reported vSSL results are obtained with this value for the MS-CMRSeg dataset and $\rho = 5000$ for the CHAOS dataset, unless otherwise stated. The difference in value of ρ reflects the differences in volume size.

5.3.4. Influence of steepness parameter

The steepness of the sigmoid function controls how soft the threshold operation performed is. If the steepness is high (harder thresholding), the class assignments of samples becomes harder, also close to the threshold. To examine the influence of the steepness parameter, κ, on the final segmentation results, we have conducted six experiments with different values of κ, from $\kappa = 0.1$ to $\kappa = 1.0$ on the MS-CMRSeg dataset[8]. The results presented in Table 6 indicate the model's robustness with respect to this parameter, and we can observe a gain of more than two percentage points in the dice score by decreasing the steepness from 1.0 to 0.5.

5.3.5. vSSL vs pSSL

To disentangle and isolate the effect from the proposed extension of the self-supervision task, we have conducted additional experiments where we train our proposed model (ADNet), and the closest competing model (ALPNet) with the two different self-supervision tasks. From the results in Table 7, we see that the supervoxels overall yield better or comparable results for both models. For our proposed ADNet, there is a significant improvement

[8] Note that this is equivalent to changing the scaling between $\alpha = 0.2$ and $\alpha = 20$ in Eq. (2).

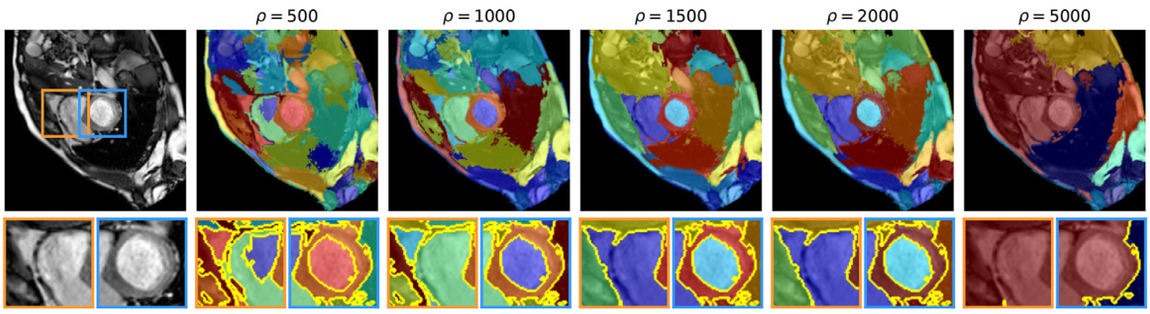


Fig. 9. Examples of supervoxel segmentation results in one slice from the MS-CMRSeg dataset for different values of ρ. The parameter ρ controls the minimum size of a supervoxel for it not to be joined with an adjacent supervoxel. A larger ρ corresponds to larger and fewer supervoxels.

Table 7
Mean dice score and standard deviation over three runs per split for ADNet and ALPNet with superpixel-based and supervoxel-based self-supevision. * indicates that the increase in mean dice score for the best performing model is statistically significant ($p < 0.05$).

Model	pSSL	vSSL	Cardiac MRI				Abdominal MRI				
			LV-BP	LV-MYO	RV	Mean	L kid.	R kid.	Spleen	Liver	Mean
ALPNet	✓		**80.65 ± 3.93**	53.31±6.31	**69.25 ± 2.80**	**67.74 ± 11.21**	56.42±5.74	50.37±7.77	44.70±7.77	56.73±3.07	52.05±4.94
		✓	79.44±2.79	**57.64 ± 3.96**	61.22±4.22	66.10±9.59	**68.19 ± 12.30**	**82.45 ± 6.27**	**55.39 ± 10.09**	**66.38 ± 5.20**	**68.10 ± 9.62***
ADNet	✓		78.25±9.68	54.59±6.42	66.37±4.73	66.40±12.07	49.65±7.59	59.00±13.77	52.47±9.56	54.78±3.87	53.97±10.00
		✓	**82.81 ± 3.20**	**59.46 ± 2.97**	**66.58 ± 4.74**	**69.62 ± 9.77***	**62.33 ± 9.70**	**86.46 ± 2.74**	**63.73 ± 11.66**	**77.12 ± 3.41**	**72.41 ± 9.96***

($p < 0.05$) in dice score from pSSL to vSSL for both datasets. Moreover, the improvements appear most prominent for the abdominal dataset, which is assumed to be related to the nature of the image volumes: In the abdominal dataset, the image volumes contain more slices and more potential information to utilize when the self-supervision task is extended to 3D, compared to the cardiac dataset.

A different implication of the proposed extension to supervoxel-based self-supervision is the enabling of training 3D CNNs for direct volume segmentation, as discussed in the next section.

5.4. Extension to one-step volume segmentation

Thus far, we have adopted a hybrid strategy to 3D segmentation, following Ouyang et al. (2020), where the 3D image volumes are segmented slice by slice, independently. However, a natural extension that is facilitated by the new self-supervision task is to adopt a 3D CNN as backbone to process the volumes in one step, thereby fully exploiting the potentially useful information along the third axis. Unfortunately, the high memory consumption and computational cost of 3D CNNs has limited their use to smaller images (in number of voxels), often obtained by downsampling the original images (Çiçek et al., 2016) or by patch-based approaches (Huo et al., 2019).

To investigate the potential of utilizing 3D convolutions to do one-step 3D segmentations within our proposed framework, we employ a 3D ResNeXt-101 (Hara et al., 2018), which is the 3D extension of ResNeXt (Xie et al., 2017), pretrained on the Kinetics-600 dataset (Kay et al., 2017), as our encoder network. The 3D ResNeXt-101 is a more resource efficient network, compared to the 3D ResNet-101, with approximately half as many trainable parameters in total. The number of parameters is comparable to the 2D ResNet-101 (see Table 8).

To retain the same spatial resolution in the embedding space as for our 2D backbone, we modify the network by *i*) removing the maxpooling in z-direction and *ii*) changing the strides in conv 3, conv 4, and conv 5 to $(1, 2, 2)$, $(1, 1, 1)$, and $(1, 1, 1)$, respectively (see architecture details in Table 9). Similarly to the 2D ResNet-101, we replace the classifier with $1 \times 1 \times 1$ convolutions to reduce the feature dimension from 2048 to 256. Each voxel is repeated three times along the channel dimension in the input to fit into the pre-trained network. The network is trained self-supervised end-to-end on 3D patches of size $(10, 215, 215)$, and the loss is optimized according to Section 4.3. During inference, we evaluate the performances under EP2 with two different levels of supervision: *i*) Only labeling the middle slice of the target class in the support volume ($k = one$), as is done in the 2D experiments. *ii*) Labeling all the support slices ($k = all$) and computing one prototype for the entire support volume, which is enabled by the volume-wise embedding.

Table 8 provides a summary of the performance of vSSL-ADNet with 3D ResNeXt-101 and 2D ResNet-101 backbones. Though it is difficult to directly compare 2D CNNs and 3D CNNs for many different reasons, such as difference in pre-training datasets and the number of weights modelling relations within slices and between slices, the results are meant to indicate the potential of using 3D convolutions in our framework to perform one-step 3D segmentation.

From the results on the cardiac dataset, we see that the differences between 2D and 3D are relatively small, which agrees with observations in previous work (Vesal et al., 2019). In the abdominal dataset, on the other hand, there appears to be a greater potential for utilizing the 3D structure via 3D convolutions. This mirrors our results from Section 5.3.5, where we found that the abdominal dataset benefited more from extending the self-supervision task from superpixels to supervoxels.

The largest performance difference between the backbones can be observed for the left kidney and spleen classes. While the 2D CNN results in a segmentation where these classes are confused, the 3D CNN leads to a better separation between the classes, as illustrated in Fig. 10. We further observe a drop in performance on the right kidney class for the 3D CNN with $k = 1$, which demonstrates the importance of having good support features to achieve robust results with the 3D backbone.

6. Limitations and outlook

The key observation leading to our anomaly-detection inspired few-shot medical image segmentation is that the foreground class typically is relatively homogeneous. By only modeling the foreground class with a single prototype, we avoid having to model

Table 8
Mean dice score and standard deviation over three runs per split for vSSL-ADNet with 2D ResNet-101 as backbone and 3D ResNeXt-101 as backbone (under EP2). * indicates that the increase in mean dice score for the best performing model is statistically significant ($p < 0.05$).

Backbone	Params	Labeled slices, k	Cardiac MRI				Abdominal MRI				
			LV-BP	LV-MYO	RV	Mean	L kid.	R kid.	Spleen	Liver	Mean
2D ResNet-101	42.50M	One	82.81±3.20	**59.46 ± 2.97**	66.58±4.74	**69.62 ± 9.77**	62.33±9.70	**86.46 ± 2.74**	63.73±11.66	77.12±3.41	72.41±9.96
3D ResNeXt-101	47.52M	One	81.28±2.51	56.47±0.75	66.22±4.24	67.99±10.60	77.95±16.57	73.55±28.69	75.04±8.55	75.48±8.58	75.50±17.71
3D ResNeXt-101	47.52M	All	**82.87 ± 1.15**	56.30±0.76	**67.93 ± 4.04**	69.03±11.15	**81.06 ± 4.20**	84.88±4.22	**75.18 ± 8.40**	**77.17 ± 8.60**	**79.58 ± 7.67***

Table 9
Modified 3D ResNeXt-101 architecture with cardinality $C = 32$ used as backbone in the 3D experiments.

Layer name	Output size	Architecture
conv 1	$(1, \frac{1}{2}, \frac{1}{2})$	$7 \times 7 \times 7$, 64, stride 1, 2, 2
conv 2	$(1, \frac{1}{4}, \frac{1}{4})$	$1 \times 3 \times 3$ max pool, stride 1, 2, 2 $\begin{bmatrix} 1 \times 1 \times 1, 128, \text{stride } 1, 1, 1 \\ 3 \times 3 \times 3, 128, \text{stride } 1, 1, 1, C = 32 \\ 1 \times 1 \times 1, 256, \text{stride } 1, 1, 1 \end{bmatrix} \times 3$
conv 3	$(1, \frac{1}{8}, \frac{1}{8})$	$\begin{bmatrix} 1 \times 1 \times 1, 256, \text{stride } 1, 1, 1 \\ 3 \times 3 \times 3, 128, \text{stride } 1, 2, 2, C = 32 \\ 1 \times 1 \times 1, 512, \text{stride } 1, 1, 1 \end{bmatrix} \times 4$
conv 4	$(1, \frac{1}{8}, \frac{1}{8})$	$\begin{bmatrix} 1 \times 1 \times 1, 512, \text{stride } 1, 1, 1 \\ 3 \times 3 \times 3, 512, \text{stride } 1, 1, 1, C = 32 \\ 1 \times 1 \times 1, 1024, \text{stride } 1, 1, 1 \end{bmatrix} \times 23$
conv 5	$(1, \frac{1}{8}, \frac{1}{8})$	$\begin{bmatrix} 1 \times 1 \times 1, 1024, \text{stride } 1, 1, 1 \\ 3 \times 3 \times 3, 1024, \text{stride } 1, 1, 1, C = 32 \\ 1 \times 1 \times 1, 2048, \text{stride } 1, 1, 1 \end{bmatrix} \times 3$
conv 6	$(1, \frac{1}{8}, \frac{1}{8})$	$1 \times 1 \times 1$, 256, stride 1, 1, 1

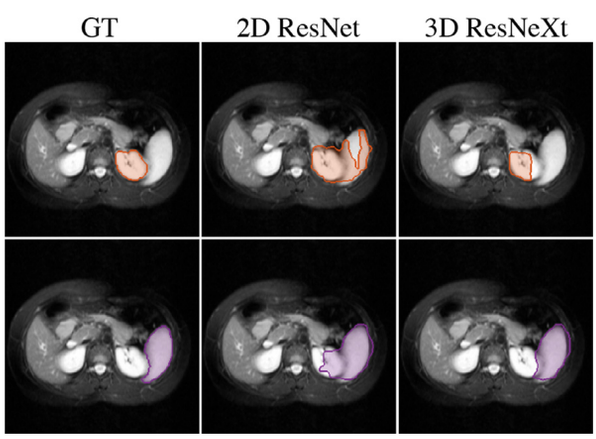


Fig. 10. Comparison of the segmentation results for the left kidney (orange, top) and spleen (purple, bottom) classes for vSSL-ADNet with 2D ResNet-101 and 3D ResNeXt-101 as backbone. The 3D CNN leads to a better separation between the classes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the large and highly inhomogeneous background, which we believe is the main challenge in prototypical few-shot medical image segmentation. However, if our assumption of a relatively homogeneous foreground class is not met, and the foreground consists of multiple distinct regions with strong edges, e.g. combining left-ventricle blood pool and left-ventricle myocardium into one foreground class (left-ventricle), modeling the foreground with one prototype might not be sufficient. This is related to the nature of the supervoxels, which tend to follow the boundaries of the structures in the image; Left-ventricle blood pool and left-ventricle myocardium will typically belong to different supervoxels during training and the network therefore learns to separate their feature representations into different clusters. To be able to capture this combined foreground class during inference, one option could be to take inspiration from PPNet (Liu et al., 2020b) and cluster the features into multiple foreground prototypes and then merge the results.

Both the superpixel-based and the supervoxel-based self-supervision tasks are inevitably vulnerable to merging different classes during training *if* the boundaries between them are weak: If the boundaries are weak, the classes will end up in the same superpixel/voxel and the network learns to embed the classes into the same cluster, which makes them difficult to separate during inference. Moreover, in the supervoxel case, it is enough for *one* slice to contain a weak boundary between the classes before they leak into the same supervoxel. This is something that happens between the left-kidney and the spleen in the abdominal dataset, and leads to confusion between these two classes during inference, thereby resulting in lower dice scores and high standard deviations. Taking into account this weak/noisy nature of the supervoxel pseudo-labels is a promising direction for future research.

7. Conclusion

In this work, we proposed a novel and end-to-end trainable anomaly detection-inspired FSS network for medical image segmentation. By approaching the segmentation task as an anomaly detection problem, our model eliminates the need to explicitly model the large and heterogeneous background class. Moreover, to train the model in an unsupervised manner, we introduced a new self-supervision task that captures the 3D nature of the data by utilizing supervoxels. We assessed our proposed model on representative datasets for cardiac segmentation and abdominal organ segmentation, and showed that it improves segmentation performance and robustness, especially in the realistic scenario where no weak labels for the query images are assumed. Furthermore, we demonstrated how the proposed model, together with the new self-supervision task, has the potential to perform one-step 3D segmentation of the entire image volumes. We believe that fully exploiting the 3D nature of the medical images in this manner for few-shot segmentation represents an interesting line of research for future work.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Stine Hansen: Conceptualization, Methodology, Software, Writing – original draft, Formal analysis. **Srishti Gautam:** Conceptualization, Methodology, Writing – review & editing. **Robert Jenssen:**

Conceptualization, Methodology, Writing – review & editing, Supervision. **Michael Kampffmeyer:** Conceptualization, Methodology, Writing – review & editing, Supervision.

Acknowledgements

This work was supported by The Research Council of Norway (RCN), through its Centre for Research-based Innovation funding scheme [grant number 309439] and Consortium Partners; RCN FRIPRO [grant number 315029]; RCN IKTPLUSS [grant number 303514]; and the UiT Thematic Initiative.

References

Abdeltawab, H., Khalifa, F., Taher, F., Alghamdi, N.S., Ghazal, M., Beache, G., Mohamed, T., Keynton, R., El-Baz, A., 2020. A deep learning-based approach for automatic segmentation and quantification of the left ventricle from cardiac cine mr images. Computerized Medical Imaging and Graphics 81, 101717.

Bai, W., Chen, C., Tarroni, G., Duan, J., Guitton, F., Petersen, S.E., Guo, Y., Matthews, P.M., Rueckert, D., 2019. Self-supervised learning for cardiac mr image segmentation by anatomical position prediction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 541–549.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: a survey. ACM computing surveys (CSUR) 41 (3), 1–58.

Chen, L., Bentley, P., Mori, K., Misawa, K., Fujiwara, M., Rueckert, D., 2019. Self-supervised learning for medical image analysis using image context restoration. Med Image Anal 58, 101539.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning. PMLR, pp. 1597–1607.

Chen, X., Sun, S., Bai, N., Tang, H., Liu, Q., Yao, S., Han, K., Zhang, C., Lu, Z., Huang, Q., et al., 2021. A deep learning-based auto-segmentation system for organs-at-risk on whole-body computed tomography images for radiation therapy. Radiotherapy and Oncology.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D u-net: learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing and Computer-assisted Intervention. Springer, pp. 424–432.

Doersch, C., Gupta, A., Efros, A.A., 2015. Unsupervised visual representation learning by context prediction. In: Proceedings of the IEEE international conference on computer vision, pp. 1422–1430.

Dong, N., Kampffmeyer, M., Voiculescu, I., 2021. Self-supervised multi-task representation learning for sequential medical images. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, pp. 779–794.

Dong, N., Xing, E.P., 2018. Few-shot semantic segmentation with prototype learning. In: British Machine Vision Conference, Vol. 3. British Machine Vision Association.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient graph-based image segmentation. Int J Comput Vis 59 (2), 167–181.

Finn, C., Abbeel, P., Levine, S., 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning. Proceedings of Machine Learning Research, pp. 1126–1135.

Hara, K., Kataoka, H., Satoh, Y., 2018. Can spatiotemporal 3D CNNS retrace the history of 2D cnns and imagenet? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6546–6555.

Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141.

Huang, Q., Huang, Y., Luo, Y., Yuan, F., Li, X., 2020. Segmentation of breast ultrasound image with semantic classification of superpixels. Med Image Anal 61, 101657.

Huo, Y., Xu, Z., Xiong, Y., Aboud, K., Parvathaneni, P., Bao, S., Bermudez, C., Resnick, S.M., Cutting, L.E., Landman, B.A., 2019. 3D whole brain segmentation using spatially localized atlas network tiles. Neuroimage 194, 105–119.

Irving, B., Franklin, J.M., Papież, B.W., Anderson, E.M., Sharma, R.A., Gleeson, F.V., Brady, M., Schnabel, J.A., 2016. Pieces-of-parts for supervoxel segmentation with global context: application to dce-mri tumour delineation. Med Image Anal 32, 69–83.

Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H., 2021. Nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods 18 (2), 203–211.

Jing, L., Tian, Y., 2020. Self-supervised visual feature learning with deep neural networks: a survey. IEEE Trans Pattern Anal Mach Intell.

Kavur, A.E., Gezer, N.S., Barış, M., Aslan, S., Conze, P.-H., Groza, V., Pham, D.D., Chatterjee, S., Ernst, P., Özkan, S., Baydar, B., Lachinov, D., Han, S., Pauli, J., Isensee, F., Perkonigg, M., Sathish, R., Rajan, R., Sheet, D., Dovletov, G., Speck, O., Nürnberger, A., Maier-Hein, K.H., Bozdağı Akar, G., Ünal, G., Dicle, O., Selver, M.A., 2021. CHAOS Challenge - combined (CT-MR) healthy abdominal organ segmentation. Med Image Anal 69, 101950. doi:10.1016/j.media.2020.101950.

Kavur, A.E., Gezer, N.S., Barış, M., Şahin, Y., Özkan, S., Baydar, B., Yüksel, U., Kılıkçıer, Ç., Olut, Ş., Bozdağı Akar, G., Ünal, G., Dicle, O., Selver, M.A., 2020. Comparison of semi-automatic and deep learning based automatic methods for liver segmentation in living liver transplant donors. Diagnostic and Interventional Radiology 26, 11–21. doi:10.5152/dir.2019.19.

Kavur, A. E., Selver, M. A., Dicle, O., Barış, M., Gezer, N. S., 2019. CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation Challenge Data. 10.5281/zenodo.3362844

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al., 2017. The kinetics human action video dataset. arXiv preprint arXiv:1705.06950.

Komodakis, N., Gidaris, S., 2018. Unsupervised representation learning by predicting image rotations. In: International Conference on Learning Representations (ICLR).

Larsson, G., Maire, M., Shakhnarovich, G., 2016. Learning representations for automatic colorization. In: European Conference on Computer Vision. Springer, pp. 577–593.

Li, G., Jampani, V., Sevilla-Lara, L., Sun, D., Kim, J., Kim, J., 2021. Adaptive prototype learning and allocation for few-shot segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.-W., Heng, P.-A., 2018. H-Denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. IEEE Trans Med Imaging 37 (12), 2663–2674.

Li, Z., Zhao, W., Shi, F., Qi, L., Xie, X., Wei, Y., Ding, Z., Gao, Y., Wu, S., Liu, J., et al., 2021. A novel multiple instance learning framework for covid-19 severity assessment via data augmentation and self-supervised learning. Med Image Anal 69, 101978.

Liu, W., Zhang, C., Lin, G., Liu, F., 2020. Crnet: Cross-reference networks for few-shot segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4165–4173.

Liu, Y., Zhang, X., Zhang, S., He, X., 2020. Part-aware prototype network for few-shot semantic segmentation. In: European Conference on Computer Vision. Springer, pp. 142–158.

Lu, Q., Li, Y., Ye, C., 2021. Volumetric white matter tract segmentation with nested self-supervised learning using sequential pretext tasks. Med Image Anal 102094.

Mishra, N., Rohaninejad, M., Chen, X., Abbeel, P., 2018. A simple neural attentive meta-learner. In: International Conference on Learning Representations (ICLR).

Misra, I., Maaten, L.v.d., 2020. Self-supervised learning of pretext-invariant representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6707–6717.

Nguyen, V.N., Løkse, S., Wickstrøm, K., Kampffmeyer, M., Roverso, D., Jenssen, R., 2020. Sen: A novel feature normalization dissimilarity measure for prototypical few-shot learning networks. Springer, pp. 118–134.

Noroozi, M., Favaro, P., 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In: European Conference on Computer Vision. Springer, pp. 69–84.

Oreshkin, B.N., Rodriguez, P., Lacoste, A., 2018. Tadam: task dependent adaptive metric for improved few-shot learning. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, pp. 719–729.

Ouyang, C., Biffi, C., Chen, C., Kart, T., Qiu, H., Rueckert, D., 2020. Self-supervision with superpixels: Training few-shot medical image segmentation without annotation. In: European Conference on Computer Vision. Springer, pp. 762–780.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A., 2016. Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2536–2544.

Rakelly, K., Shelhamer, E., Darrell, T., Efros, A., Levine, S., 2018. Conditional networks for few-shot semantic segmentation. In: International Conference on Learning Representations (ICLR).

Ravi, S., Larochelle, H., 2017. Optimization as a model for few-shot learning. In: International Conference on Learning Representations (ICLR).

Ren, X., Malik, J., 2003. Learning a classification model for segmentation. In: Computer Vision, IEEE International Conference on, Vol. 2. IEEE Computer Society. 10–10

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer, pp. 234–241.

Roy, A.G., Siddiqui, S., Pölsterl, S., Navab, N., Wachinger, C., 2020. "Squeeze & excite" guided few-shot segmentation of volumetric images. Med Image Anal 59, 101587.

Ruff, L., Kauffmann, J.R., Vandermeulen, R.A., Montavon, G., Samek, W., Kloft, M., Dietterich, T.G., Müller, K.-R., 2021. A unifying review of deep and shallow anomaly detection. Proc. IEEE.

Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T., 2016. Meta-learning with memory-augmented neural networks. In: International Conference on Machine Learning, pp. 1842–1850.

Shaban, A., Bansal, S., Liu, Z., Essa, I., Boots, B., 2017. One-shot learning for semantic segmentation. In: Proceedings of the British Machine Vision Conference (BMVC), pp. 167.1–167.13.

Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems, pp. 4077–4087.

Stutz, D., Hermans, A., Leibe, B., 2018. Superpixels: an evaluation of the state-of-the-art. Comput. Vision Image Understanding 166, 1–27.

Tian, Z., Zhao, H., Shu, M., Yang, Z., Li, R., Jia, J., 2020. Prior guided feature enrichment network for few-shot segmentation. IEEE Trans Pattern Anal Mach Intell.

Tsochatzidis, L., Koutla, P., Costaridou, L., Pratikakis, I., 2021. Integrating segmentation information into cnn for breast cancer diagnosis of mammographic masses. Comput Methods Programs Biomed 200, 105913.

Vesal, S., Ravikumar, N., Maier, A., 2019. Automated multi-sequence cardiac mri segmentation using supervised domain adaptation. In: International Workshop on Statistical Atlases and Computational Models of the Heart. Springer, pp. 300–308.
Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al., 2016. Matching networks for one shot learning. In: Advances in Neural Information Processing systems, pp. 3630–3638.
Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J., 2019. Panet: Few-shot image semantic segmentation with prototype alignment. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 9197–9206.
Wilcoxon, F., 1992. Individual Comparisons by Ranking Methods. In: Breakthroughs in statistics. Springer, pp. 196–202.
Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K., 2017. Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500.
Zhang, B., Xiao, J., Qin, T., 2021. Self-guided and cross-guided learning for few-shot segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
Zhang, C., Lin, G., Liu, F., Yao, R., Shen, C., 2019. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5217–5226.
Zhang, R., Isola, P., Efros, A.A., 2016. Colorful image colorization. In: European conference on computer vision. Springer, pp. 649–666.
Zhang, X., Wei, Y., Yang, Y., Huang, T.S., 2020. Sg-one: similarity guidance network for one-shot semantic segmentation. IEEE Trans Cybern 50 (9), 3855–3865.
Zhu, J., Li, Y., Hu, Y., Ma, K., Zhou, S.K., Zheng, Y., 2020. Rubik's cube+: a self-supervised feature learning framework for 3D medical image analysis. Med Image Anal 64, 101746.
Zhuang, X., 2016. Multivariate mixture model for cardiac segmentation from multi--sequence mri. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 581–588.
Zhuang, X., 2018. Multivariate mixture model for myocardial segmentation combining multi-source images. IEEE Trans Pattern Anal Mach Intell 41 (12), 2933–2946.

Paper III

ADNet++: A few-shot learning framework for multi-class medical image volume segmentation with uncertainty-guided feature refinement

Stine Hansen, Srishti Gautam, Suaiba Amina Salahuddin, Michael Kampffmeyer, and Robert Jenssen

In submission

ADNet++: A few-shot learning framework for multi-class medical image volume segmentation with uncertainty-guided feature refinement

Stine Hansen[a,*], Srishti Gautam[a], Suaiba Amina Salahuddin[a], Michael Kampffmeyer[a], Robert Jenssen[a]

[a]*Department of Physics and Technology, UiT The Arctic University of Norway, NO-9037 Tromsø, Norway*

Abstract

A major barrier to applying deep segmentation models in the medical domain is their typical data-hungry nature, requiring experts to collect and label large amounts of data for training. As a reaction, prototypical few-shot segmentation (FSS) models have recently gained traction as data-efficient alternatives. Nevertheless, despite the recent progress of these models, they still have some essential shortcomings that must be addressed. In this work, we focus on three of these shortcomings: i) the lack of uncertainty estimation, ii) the lack of a guiding mechanism to help locate edges and encourage spatial consistency in the segmentation maps, and iii) the models' inability to do one-step multi-class segmentation. Without modifying or requiring a specific backbone architecture, we propose a modified prototype extraction module that facilitates the computation of uncertainty maps in prototypical FSS models, and show that the resulting maps are useful indicators of the model uncertainty. To improve the segmentation around boundaries and to encourage spatial consistency, we propose a novel feature refinement module that leverages structural information in the input space to help guide the segmentation in the feature space. Furthermore, we demonstrate how uncertainty maps can be used to automatically guide this feature refinement. Finally, to avoid ambiguous voxel predictions that occur when images are segmented class-by-class, we propose a procedure to perform one-step multi-class FSS. The efficiency of our proposed methodology is evaluated on two representative datasets for abdominal organ segmentation (CHAOS dataset) and cardiac segmentation (MS-CMRSeg dataset). The results show that our proposed methodology significantly (one-sided Wilcoxon signed rank test, $p < 0.05$) improves the baseline, increasing the overall dice score with +5.2 and +3.1 percentage points for the CHAOS dataset and MS-CMRSeg dataset, respectively.

Keywords:
Few-shot segmentation, Medical image segmentation, Uncertainty estimation

1. Introduction

Accurate image segmentation is an essential prerequisite for various clinical applications, such as radiotherapy treatment planning (Gonzalez et al., 2021), tissue quantification (Militello

*Corresponding author

et al., 2019), and diagnostics (Tsochatzidis et al., 2021). Prototypical few-shot segmentation (FSS) models have recently shown promise as data efficient alternatives to solving this task (Tang et al., 2021; Yu et al., 2021; Ouyang et al., 2022; Hansen et al., 2022), eliminating the need to collect and annotate large amounts of images, which is a key challenge for the application of deep learning models in the medical domain (Shen et al., 2020). In particular, Hansen et al. (2022) propose ADNet, an anomaly detection-inspired approach to FSS that simplifies the problem by refraining from explicitly modeling the difficult background class. This results in a model that is robust to the large and inhomogeneous background class, thus for the first time enabling one-step volume-wise prototypical FSS, yielding state-of-the-art performance.

When trained, the FSS models mentioned above can generalize from a *few* labeled samples to solve new segmentation tasks during inference. Specifically, a few labeled examples are exploited to extract class-wise prototypes that are used to make predictions on the unlabeled test data. However, despite their recent advances, current FSS models have some fundamental shortcomings that need to be addressed to approach clinical application.

Firstly, existing medical FSS models do *not* provide any measure of uncertainty for their predictions, which limits their trustworthiness. Knowing when the model is uncertain and therefore more likely to make mistakes is important information that should accompany the prediction in a safety-critical application such as medical image segmentation (Kompa et al., 2021).

Secondly, in current methods, the segmentation is performed directly on the spatially compressed feature representation, without any mechanism to guide the precise location of edges and structures in the image. The final segmentation map is simply obtained by re-sampling the output via bi-/tri-linear up-sampling, resulting in segmentation masks that typically struggle to accurately locate edges.

Finally, in medical image segmentation, there are often *multiple* foreground classes of relevance, e.g. a number of different organs. However, current medical FSS methods only support *binary* foreground/background segmentation and are forced to segment the images class-by-class. In addition to unnecessary forward passes, this can lead to regions with ambiguous predictions as voxels might get classified as "foreground" for multiple classes.

In this work, we focus on the inference phase to address the above-mentioned shortcomings. Without requiring modification or re-training of the network parameters, we develop methods to better exploit the available information in order to provide more *trustworthy* and more *accurate* predictions. Specifically, to facilitate the computation of uncertainty maps in prototypical FSS models we propose a modified prototype extraction module that introduces a Bernoulli distributed variable for each voxel location in the feature representation. Uncertainty maps are then based on the predictive distribution estimated from a set of prototypes extracted by this proposed module. Further, to alleviate the loss of spatial details and encourage spatial consistency in the predictions, we propose a novel feature refinement module that leverages supervoxels in the inference phase. Supervoxels are collections of voxels that represent compact regions of coherent voxel intensities and/or textures in the image volume. By utilizing supervoxels, we are able to encourage spatial consistency in the prediction, and help locate edges accurately in the segmentation map. Additionally, we show how uncertainty maps can be used to automatically guide this feature refinement. Finally, to avoid the problem of ambiguous voxel predictions, we propose a procedure to perform one-step multi-class FSS.

Exploiting its ability to perform volume-wise one-step FSS, we illustrate the benefit of the proposed methodology in the context of the current state-of-the-art 3D medical FSS model, ADNet (Hansen et al., 2022), and refer to the modified model as ADNet++.

To summarize, our contributions are as follows:

1. We propose a novel prototype extraction module that, with negligible computational overhead, can produce uncertainty maps for prototypical FSS models.
2. We propose a novel feature refinement module that leverages supervoxels to encourage spatial consistency and to locate edges in the segmentation masks. We also show how uncertainty maps can be used to guide the feature refinement.
3. We propose a one-step multi-class segmentation procedure to avoid ambiguous voxel predictions.

2. Related work

2.1. Medical few-shot segmentation

Lately, few-shot learning models have demonstrated promising segmentation performance on medical images (Roy et al., 2020; Tang et al., 2021; Yu et al., 2021; Ouyang et al., 2022; Hansen et al., 2022). Previous works can be categorized into methods that require labeled data during the training phase (Roy et al., 2020; Tang et al., 2021; Yu et al., 2021) and methods that are trained self-supervised on unlabeled data (Ouyang et al., 2022; Hansen et al., 2022). In the former category, as the first medical FSS model, Roy et al. (2020) propose a two-branched architecture, where the support features are used to implicitly guide the query segmentation through multiple interaction blocks. The succeeding works build on prototypical ideas (Snell et al., 2017), with a direct comparison between the query features and computed support prototypes. In (Yu et al., 2021), the authors propose a prototype network that leverages strong spatial priors by dividing the input images into grids and solving the segmentation problem for each grid-element separately via multiple local prototypes. Tang et al. (2021) propose a prototype network with a recurrent mask refinement, where the previous query prediction is used to refine the query features in an iterative manner.

The few-shot learning models discussed above are only few-shot in the sense that a *trained* few-shot model only needs a few labeled instances to segment a new class. During the training phase, the models still require abundant labeled data in order to avoid over-fitting. However, the availability of labeled data is often limited in the medical setting, and to overcome this challenge, Ouyang et al. (2022) propose a self-supervised few-shot segmentation model. The network itself, ALPNet, is a prototype based network that introduce adaptive local prototype pooling where local prototypes are computed on a regular grid to preserve local information. As opposed to Yu et al. (2021), Ouyang et al. (2022) do not divide the input images into grids, but segment the images as one segmentation problem. To train the network, Ouyang et al. (2022) propose a new self-supervision task for segmentation by utilizing superpixels. The authors construct a pseudo-labeled support/query pair based on *one* unlabeled image slice and its unsupervised superpixel segmentation. By sampling one random superpixel, they binarize the superpixel segmentation and consider this the support label belonging to the image slice. Then the query image and label are created by applying random spatial and intensity transformations to the support image-label pair. Hansen et al. (2022) build further on this work and extend the self-supervision task to supervoxels, utilizing the 3D information in the image volumes. Further, they propose an anomaly detection-inspired prototypical segmentation network, ADNet, where they avoid modeling the large and inhomogeneous background class with prototypes. While previous methods are limited to slice-by-slice segmentation of the image volumes, Hansen et al. (2022) are the first to extend prototypical FSS to one-step volume-wise 3D segmentation.

A drawback of all the methods discussed above is that they only perform binary image segmentation and are forced to segment multi-class segmentation problems in a class-by-class manner. Further, due to the loss of spatial detail during the encoding of the images, the models have difficulty with accurately locating edges. Finally, these models do not provide any measure of uncertainty of their predictions, which is important to build trustworthy models. In this work, we build further on the branch of self-supervised models and propose a framework for one-step multi-class medical image segmentation that provides uncertainty maps to accompany the model predictions and that involves a feature refinement that addresses the loss of spatial detail during encoding.

2.2. Uncertainty estimation

In critical decision-making processes, such as medical image segmentation, there is a need to quantify model uncertainty. That is, in addition to the model prediction, a measure of model uncertainty should be conveyed to the user to improve both safety and the reliability of the model (Kompa et al., 2021).

In medical image segmentation, Bayesian approximation (Gal and Ghahramani, 2016) and ensemble learning techniques (Lakshminarayanan et al., 2017) are often used for uncertainty quantification (Karimi et al., 2019; Wickstrøm et al., 2020; Harper and Southern, 2020; van Hespen et al., 2021). While ensemble approaches are conceptually simpler, they typically require training of multiple models, making them computationally expensive.

In few-shot segmentation outside the medical domain, Johnander et al. (2021) propose a few-shot learner formulated as a deep Gaussian process. The Gaussian process works as a layer in the network that predicts the mean and covariance of the conditional probability distribution of the query mask given the query image and support set. This information is then fed to a decoder that produces the final output. The model is thus able to model the uncertainty and uses the information to improve the segmentation performance. Concurrently, Kim et al. (2021) propose another Gaussian process inspired technique to few-shot segmentation by using a network to estimate the uncertainty. They then use the uncertainty maps to exclude samples with high prediction uncertainty for pseudo label construction in a semi-supervised setting. While these approaches provide uncertainty maps in the FSS setting, they are model-specific and thus not directly applicable to the current state-of-the-art medical FSS models, raising the need for architecture-agnostic approaches.

3. Problem definition

Given a training dataset with classes C_{train}, the goal of FSS is to obtain a model that, based on only a few labeled samples, can learn to segment the target classes C_{test}. The model is trained and tested in episodes, where a support set consisting of k labeled support images is used to predict the segmentation of N classes in the unlabeled query image. The support set is defined as $\mathcal{S} = \left\{(\mathbf{X}_1^s, \mathbf{Y}_1^s), ..., (\mathbf{X}_k^s, \mathbf{Y}_k^s))\right\}$ and the query set as $\mathcal{Q} = \{\mathbf{X}^q\}$, where $\mathbf{X}^* \in \mathbb{R}^{C\times H\times W}$ represents the image volumes and $\mathbf{Y}^* \in \mathbb{R}^{C\times H\times W}$ the corresponding voxel-wise annotations[1].

[1] Superscript * denotes support (s) or query (q).

4. Methods

4.1. Multi-class anomaly detection-inspired segmentation

As demonstrated in (Hansen et al., 2022), an anomaly detection-inspired approach to few-shot medical image segmentation results in a model that is less sensitive to variations in the background class, thus enabling one-step volume-wise 3D segmentation (as opposed to slice-by-slice 2D segmentation). As a consequence, this framework facilitates the extraction of all class-prototypes simultaneously, thereby making it suitable for multi-class segmentation.

Similar to the original ADNet, ADNet++ uses a backbone network f_θ to encode the support images $\left\{\mathbf{X}_i^s\right\}_{i=1}^k$ and query images $\left\{\mathbf{X}_i^q\right\}_{i=1}^k$ into deep feature maps, $\mathbf{F}_i^s = f_\theta(\mathbf{X}_i^s)$ and $\mathbf{F}_i^q = f_\theta(\mathbf{X}_i^q)$, respectively. Due to max-pooling operations and strided convolutions in the backbone network, the spatial resolution of these feature maps is compressed, compared to the input. The support feature map is therefore up-sampled to original size (C, H, W) before computing the class-specific prototypes through masked average pooling. Let $\Omega = \{\mathbf{r}_i\}_{i=1}^{C\cdot H\cdot W}$ denote the set of all voxel positions $\mathbf{r} = (x, y, z)$ in the image. Prototype $\mathbf{p}_c \in \mathbb{R}^d$, representing class c, is defined as:

$$\mathbf{p}_c = \frac{\sum_{i=1}^k \sum_{\mathbf{r}\in\mathbf{\Omega}} \mathbf{F}_i^s(\mathbf{r}) \cdot \mathbf{Y}_c^s(\mathbf{r})}{\sum_{i=1}^k \sum_{\mathbf{r}\in\mathbf{\Omega}} \mathbf{Y}_c^s(\mathbf{r})}, \tag{1}$$

where $\mathbf{Y}_c^s = \mathbb{1}(\mathbf{Y}_i^s = c)$ is the ground-truth mask of class c. Unlike ADNet, which only performs binary segmentation and thus only extracts *one* class-prototype at a time, we propose a procedure to perform one-step multi-class segmentation. In a N-class segmentation problem, this results in a set of N prototypes $\mathcal{P} = \{\mathbf{p}_c\}_{c=1}^N$, for which we compute a set of N anomaly score maps $\mathcal{S} = \{\mathbf{S}_c\}_{c=1}^N$, computed as:

$$\mathbf{S}_c(\mathbf{r}) = -\alpha \cos\left(\mathbf{F}^q(\mathbf{r}), \mathbf{p}_c\right), \tag{2}$$

where $\alpha = 20$ is a commonly used scaling factor (Wang et al., 2019; Ouyang et al., 2022; Hansen et al., 2022). The resulting anomaly score maps represent the dis-similarity between each voxel feature vector $\mathbf{F}^q(\mathbf{r})$ and each of the class-prototypes in $\mathcal{P}$. The soft foreground predictions for each foreground class $c = 1, \ldots, N$ are then found by thresholding the anomaly score maps with a learned threshold T:

$$\hat{\mathbf{Y}}_c^q(\mathbf{r}) = 1 - \sigma(\mathbf{S}_c(\mathbf{r}) - T), \tag{3}$$

where σ is the Sigmoid function. For a general number of N foreground classes, the soft background mask is then computed as:

$$\hat{\mathbf{Y}}_{c=0}^q(\mathbf{r}) = 1 - \max\left\{\hat{\mathbf{Y}}_c^q(\mathbf{r}) : c = 1, \ldots, N\right\}. \tag{4}$$

Finally, if $N > 1$ the class probabilities are obtained by scaling the scores with a softmax function. This assures that no voxel can be assigned to more than one class, thereby preventing the ambiguous voxel predictions in binary class-by-class segmentation, occurring when a voxel lies within the threshold of multiple class-prototypes.

The network is then trained as in (Hansen et al., 2022), in an end-to-end manner to optimize a loss function consisting of three terms:

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_T + \mathcal{L}_{PAR}, \tag{5}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss between the query prediction and the query label:

$$\mathcal{L}_{CE} = -\frac{1}{|\Omega|} \sum_{\mathbf{r}\in\mathbf{\Omega}} \sum_{c=0}^{N} \hat{\mathbf{Y}}_c^q(\mathbf{r}) \log \mathbf{Y}_c^q(\mathbf{r}), \tag{6}$$

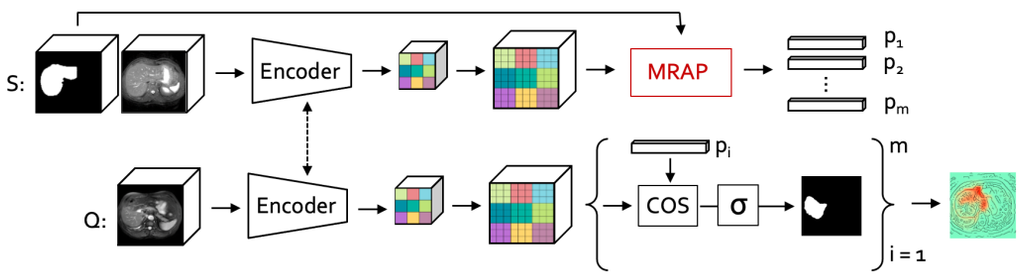


Figure 1: We facilitate the estimation of uncertainty maps in prototypical FSS models by replacing the deterministic masked average pooling module with a randomized alternative, MRAP, denoted in red. This allows us to generate a set of prototypes, and thereby a set of query predictions that can be used to estimate the model uncertainty.

where $|\cdot|$ indicates the cardinality of the set, $\mathcal{L}_T = T/\alpha$ is a loss on the threshold to encourage a compact embedding of the foreground classes, and $\mathcal{L}_{PAR}$ is the prototype alignment regularization loss from (Wang et al., 2019), obtained by reversing the roles of the support and query. The predicted query mask is used to segment the support image, and the loss is computed as the cross-entropy loss between the predicted support mask and the support ground-truth mask:

$$\mathcal{L}_{PAR} = -\frac{1}{|\Omega_s|} \sum_{\mathbf{r}\in\Omega_s} \sum_{c=0}^{N} \hat{\mathbf{Y}}_c^s(\mathbf{r}) \log \mathbf{Y}_c^s(\mathbf{r}), \tag{7}$$

where Ω_s is the set of voxel positions in the support image.

After the model is trained, the weights (θ, T) are frozen and the inference episodes are sampled from C_{test}.

4.2. Uncertainty estimation

To obtain a measure of uncertainty for the model's predictions, we take inspiration from Gal and Ghahramani (2016), who exploit dropout layers in the network architecture to be able to represent the model uncertainty. As illustrated in Figure 1, we suggest an architecture-agnostic approach to generate uncertainty maps by randomizing the masked average pooling during prototype generation in Equation 1. Instead of applying deterministic masked average pooling to obtain one prototype per class, we propose to perform masked randomized average pooling (MRAP) to obtain a set of m prototypes per class c $\mathcal{P}_c = \left\{\boldsymbol{p}_j\right\}_{j=1}^{m}$ from the support set as:

$$\mathbf{p}_j = \frac{\sum_{i=1}^{k} \sum_{\mathbf{r}\in\Omega} \mathbf{F}_i^s(\mathbf{r}) \cdot \mathbf{Y}_c^s(\mathbf{r}) \cdot \mathbf{M}_i(\mathbf{r})}{\sum_{i=1}^{k} \sum_{\mathbf{r}\in\Omega} \mathbf{Y}_c^s(\mathbf{r}) \cdot \mathbf{M}_i(\mathbf{r})}, \tag{8}$$

where $\mathbf{M}_i(\mathbf{r})$ is sampled from a Bernoulli(ρ) distribution. ρ is the probability of $\mathbf{M}_i(\mathbf{r})$ taking the value one and is set to 0.5. From this set of prototypes, we can obtain a set of anomaly scores $\{\mathbf{S}_i\}_{i=1}^{m}$, and thereby predictions $\left\{\hat{\mathbf{Y}}_i^q\right\}_{i=1}^{m}$ for the query image. These predictions can be considered samples from an approximate predictive distribution, and the model uncertainty map can be estimated as the predictive entropy (Gal, 2016). Therefore, by computing the voxel-wise predictive entropy of the m predictions, we obtain the uncertainty map as:

$$\mathbf{U}(\mathbf{r}) = -\sum_{c} \bar{\mathbf{Y}}_c(\mathbf{r}) \log \bar{\mathbf{Y}}_c(\mathbf{r}), \tag{9}$$

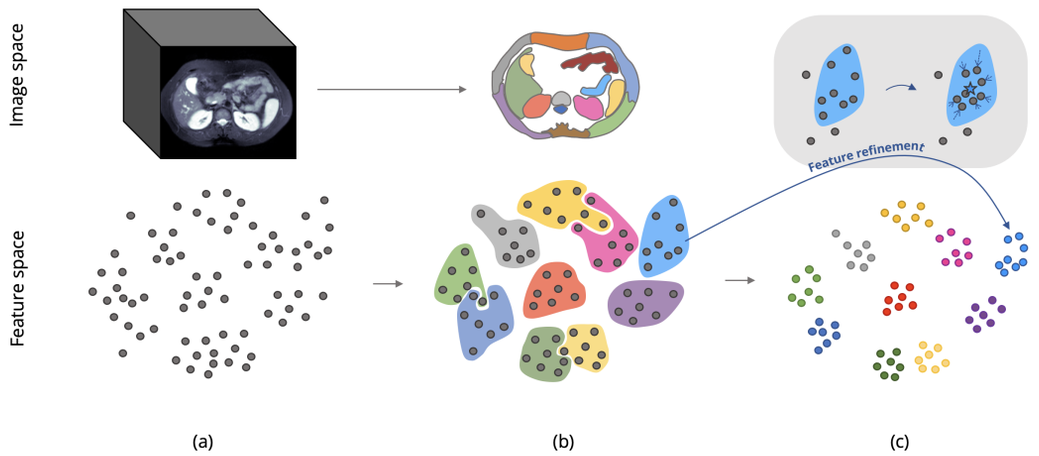


Figure 2: Conceptual illustration of the feature refinement process. (a) In the encoding process, the input image is transformed into a set of feature vectors (grey dots). (b) Supervoxels are generated in the input space and can thus be used to locate feature vectors that "belong" together in the input space. (c) The refinement process consists in moving the feature vectors within the supervoxel towards its center (indicated with blue star), leading to a more compact embedding where the edges defined in the input space are respected.

where $\bar{\mathbf{Y}}_c = \frac{1}{m}\sum_{j=1}^{m}\hat{\mathbf{Y}}_j^q$ is the average (soft) prediction map of class c. These uncertainty maps can be used to visualize and assess the voxel-wise uncertainty of the model's predictions. Further, in the next section, we show how these uncertainty maps can be leveraged to guide the proposed feature refinement.

4.3. Supervoxel-informed feature refinement module

Assuming that supervoxels capture voxels that belong together in the input space, it follows that a segmentation model should assign consistent class labels for all voxels within the same supervoxel. To encourage this spatial consistency, we propose a supervoxel-informed feature refinement module that refines the embedded image representations to respect edges as defined by the supervoxels. If a supervoxel defines a set of voxels that belong together in the input space, it consequently also defines a set of feature vectors that should belong together in the feature space, and as the encoding of images involves a spatial compression with loss of spatial details, the supervoxel-informed refinement can thus act as a mechanism to guide the precise location of edges and structures in the output. The concept of the proposed supervoxel-informed feature refinement (SFR) module is illustrated in Figure 2.

To refine the query features during inference, the up-sampled feature maps are refined as follows. Each query image $\mathbf{x}^q$ is clustered into a set of M non-overlapping supervoxels $\pi = \{\pi_1, ..., \pi_M\}$, representing homogeneous regions in the input image. Overlaying this supervoxel segmentation on top of the up-sampled query feature map, each supervoxel π_i defines a set of voxel feature vectors, corresponding to a homogeneous region in the input image. For a feature vector $\mathbf{F}^q(\mathbf{r}) \in \pi_i$, we propose a refined voxel feature vector $\mathbf{F}^q(\mathbf{r})'$, computed as:

$$\mathbf{F}^q(\mathbf{r})' = \beta\mathbf{F}^q(\mathbf{r}) + (1-\beta)\boldsymbol{\mu}_i, \tag{10}$$

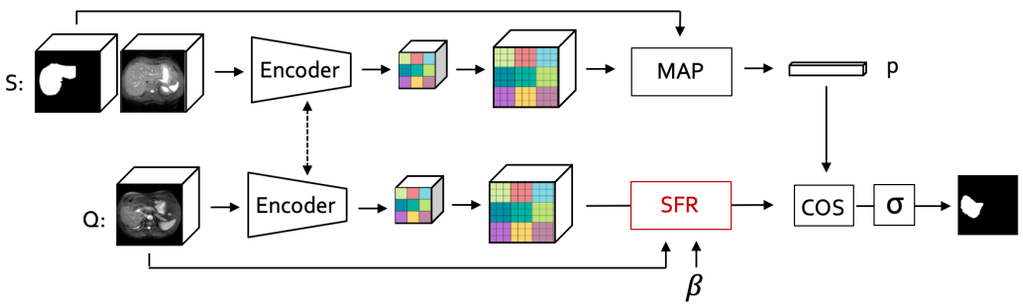


Figure 3: Workflow of the proposed feature refinement module. The module acts to refine the features before entering the classifier. The original features, the input image and a choice of β is input to the module. The refined features then follow the ordinary pipeline to produce the end segmentation result.

where $\boldsymbol{\mu}_i$ is the center of π_i, given by:

$$\boldsymbol{\mu}_i = \frac{1}{|\pi_i|} \sum_{\mathbf{F}^q(\mathbf{r}) \in \pi_i} \mathbf{F}^q(\mathbf{r}), \tag{11}$$

and β is a refinement parameter controlling the size of the feature vectors' movement, ranging from $\beta = 1$ with no movement to $\beta = 0$ where the feature vector moves all the way to its supervoxel center. However, choosing β in this way, as a fixed constant for all voxels, is quite restrictive. A dynamic $\beta(\mathbf{r})$, on the other hand, would increase the module's flexibility by allowing different regions in the feature map to experience different degrees of refinement. One possible approach to obtain a dynamic refinement is to utilize the uncertainty map:

$$\beta(\mathbf{r}) = 1 - \mathbf{U}(\mathbf{r}). \tag{12}$$

In this way, uncertain voxels, typically on the boundaries between classes, get a lower β and rely more on the sharp edge information in the supervoxels, and vice versa. Exploiting the uncertainty map has the additional advantage that no labeled data is required to determine β[2]. Figure 3 illustrates the module in the FSS framework.

Ultimately, the feature refinement module is determined by two parameters: i) the number of supervoxels M (or effectively the supervoxel *size*) and ii) the feature refinement parameter, β. The choice of these parameters is explored in Section 6.4.2.

5. Supervoxel generation

Supervoxels are computed offline for all the query images using a 3D extension[3] of the Felzenszwalb's efficient graph-based segmentation algorithm (Felzenszwalb and Huttenlocher, 2004). This is the same algorithm that is used to generate pseudo-labels for the self-supervised training in (Ouyang et al., 2022) and (Hansen et al., 2022), and is known to produce superpixels with irregular shapes and sizes that adhere well to image boundaries (Achanta et al., 2012).

The algorithm has a parameter controlling the minimum supervoxel size, and effect of this parameter on the final segmentation result is explored in Section 6.4.2.

[2]Determination of the "ideal" fixed β requires a line-search on an annotated validation set.

[3]`https://github.com/sha168/Felzenszwalb-supervoxel-segmentation`.

6. Experiments

6.1. Experiment setup

Datasets. We demonstrate the properties and performance of the proposed ADNet++ by conducting experiments on two publicly available benchmark datasets[4] in medical image segmentation: i) the bSSFP fold from the Multi-sequence Cardiac MRI Segmentation (**MS-CMRSeg**) challenge from MICCAI 2019 (Zhuang, 2016, 2018), and ii) task 5 from the Combined Healthy Abdominal Organ Segmentation (**CHAOS**) Challenge from ISBI 2019 (Kavur et al., 2019, 2020, 2021). Both datasets consist of volumetric magnetic resonance imaging (MRI) scans. The MS-CMRSeg dataset contains 20 cardiac MRIs with ground-truth segmentations for left-ventricle blood pool (LV-BP), left-ventricle myocardium (LV-MYO), and right ventricle (RV), whereas the CHAOS dataset contains 20 T2-SPIR MRIs with ground-truth segmentations for left kidney (L. kid.), right kidney (R. kid.), spleen, and liver.

Prior to training, the data is pre-processed following common practice (Ouyang et al., 2022; Hansen et al., 2022): First, we cut the top 0.5% intensities. Then, we re-sample and crop the image volumes such that the short-axis slices in the MS-CMRSeg dataset and the axial slices in the CHAOS dataset have the same size (256×256).

Evaluation metric. We employ the dice similarity coefficient (DSC) to compare model predictions to ground-truth segmentation masks. The DSC between a model prediction $\hat{Y}$ and the ground-truth Y is computed as:

$$\mathrm{DSC}(Y, \hat{Y}) = 2\frac{|Y \cap \hat{Y}|}{|Y| + |\hat{Y}|} \cdot 100\%. \tag{13}$$

Thus, the DSC varies from 100%, indicating perfect overlap between the segmentations, to 0%, when the segmentations have no overlap.

Baselines. To evaluate the effectiveness of our proposed method, we compare it to two baseline models for medical few-shot segmentation, ALPNet (Ouyang et al., 2022) and ADNet (Hansen et al., 2022). ALPNet is a framework designed for slice-wise segmentation of medical image volumes. This means that the model requires a scheme for support-query matching during the inference episodes. In the original paper, this was solved by assuming availability of weak label information on the query volumes during inference. In this work we do *not* assume the availability of such weak labels and follow the more realistic evaluation protocol 2 in (Hansen et al., 2022) where the middle slice in the support target volume is used to segment the entire query volume slice-by-slice. The second baseline is ADNet that performs volume-wise segmentation.

Evaluation Protocol. The trained models are evaluated in a five-fold cross-validation scheme where the test fold is held out during training. During inference, we sample all possible support/query combinations for the volumes in the fold to make the evaluation unbiased towards specific choices of support and query. In the tables, we report mean dice (with standard deviations) over all folds, where each fold is repeated thrice to account for the stochasticity in the optimization. To indicate statistically significant improvements, one-sided Wilcoxon signed rank tests (Wilcoxon, 1992) are performed to compare the mean DSC across all runs.

[4] Links to datasets: MS-CMRSeg and CHAOS.

Table 1: Quantitative comparison of the proposed method to the baseline models. Mean DSC with standard deviations are reported for three runs per fold. * indicates that the increase in mean DSC, compared to the ADNet baseline, is statistically significant ($p < 0.05$).

Method		**Abdominal MRI**				
		L. kid.	R. kid.	Spleen	Liver	Mean
ALPNet	2D	51.30 ± 11.61	47.66 ± 10.31	42.02 ± 16.71	56.12 ± 7.00	49.29 ± 5.17
ADNet	3D	79.57 ± 7.55	81.41 ± 10.17	68.03 ± 24.05	74.29 ± 23.39	75.82 ± 5.20
ADNet++		**86.80 ± 6.01**	**86.62 ± 10.37**	**75.69 ± 26.21**	**74.85 ± 23.82**	**80.99 ± 5.73***
Method		**Cardiac MRI**				
		LV-BP	LV-MYO	RV	Mean	
ALPNet	2D	81.30 ± 6.80	54.87 ± 7.30	**68.38 ± 10.67**	68.18 ± 10.79	
ADNet	3D	80.95 ± 5.50	53.68 ± 5.52	66.12 ± 10.14	66.92 ± 11.15	
ADNet++		**82.57 ± 6.55**	**60.02 ± 5.66**	67.44 ± 11.37	**70.01 ± 9.38***	

Implementation details. The implementation of ADNet++ is based on the PyTorch (v1.7.1) implementation of 3D ADNet[5], and the training phase is identical to (Hansen et al., 2022): We optimize the weights using stochastic gradient descent over 25k iterations with momentum 0.9, learning rate 1e-3, decay rate 0.98 per 1k iterations, and a weight decay of 5e-4. To account for the class imbalance, a weighted cross-entropy loss is employed, where the foreground and background weights are set to 1.0 and 0.1, respectively.

6.2. Comparison to state-of-the-art

Table 1 presents a quantitative comparison of the proposed method and the previous state-of-the-art methods. As ALPNet is designed for 2D slice-wise segmentation, it relies on weak label information to locate the query target volumes in order to achieve state-of-the-art performance (Hansen et al., 2022). Without the weak labels, this model performs poorly on the CHAOS dataset where the background is large with variations throughout the image volume. ADNet and ADNet++ are designed to handle a large, inhomogeneous background class and perform well in this setting, with ADNet++ significantly ($p < 0.05$) improving the overall DSC of ADNet by +5.2 percentage points. For the MS-CMRSeg dataset, the segmentation performances of ALPNet and ADNet are more similar, as found in (Hansen et al., 2022). Nevertheless, ADNet++ still significantly ($p < 0.05$) improves the performance of ADNet by +3.1 percentage points and the performance of ALPNet by +1.8 percentage points.

6.3. Uncertainty maps

Figure 4 visualizes four example slices from the CHAOS dataset with corresponding ground-truths, predictions, and uncertainty maps obtained by sampling $m = 10$ prototypes per class in the proposed prototype extraction module. From these examples, we can see that the model uncertainty typically is higher for voxels close to and on the boundaries between classes. Furthermore, when the model makes mistakes (e.g. where it over-segments the liver), we can see how the uncertainty map highlights these areas as uncertain.

[5] `https://github.com/sha168/ADNet`.

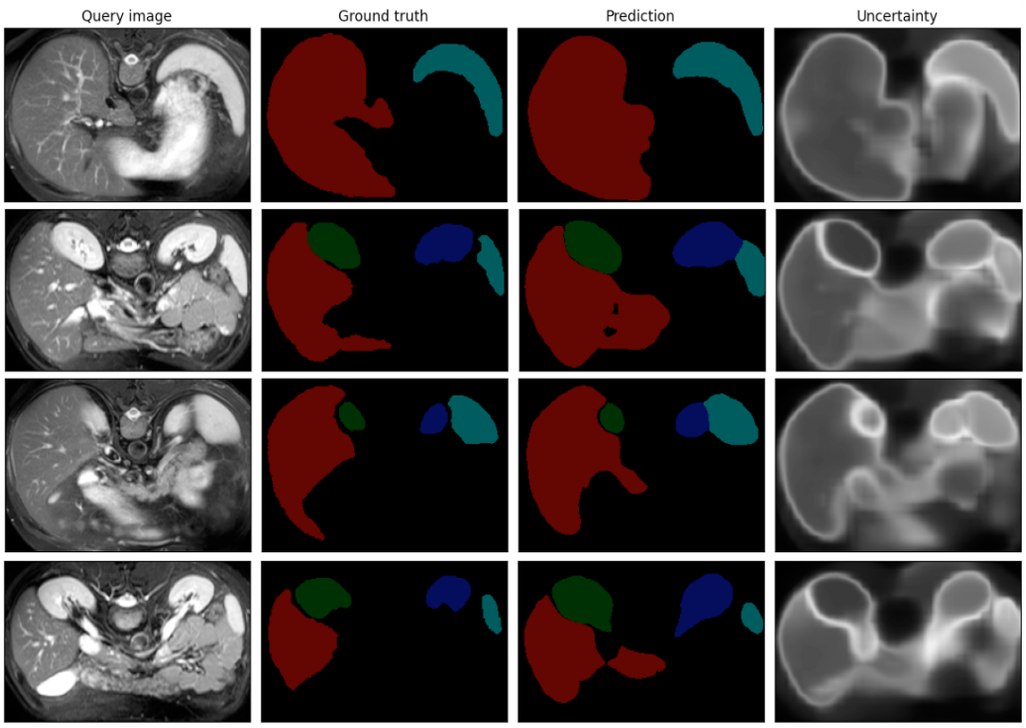


Figure 4: Illustration of example slices from the CHAOS dataset with corresponding ground-truths, predictions and uncertainty maps. The uncertainty maps typically highlight the boundary regions between classes. (red=liver, green=right kidney, dark blue=left kidney, and light blue=spleen)

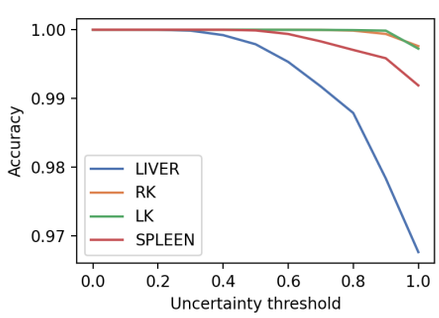


Figure 5: Relationship between accuracy and estimated uncertainty. By successively including more uncertain voxels, the segmentation accuracy decreases.

Table 2: Quantitative evaluation of the proposed components' contribution on the CHAOS dataset. Mean DSC with standard deviations are reported for three runs per fold. * indicates that the increase in mean DSC, compared to the ADNet baseline, is statistically significant ($p < 0.05$).

Method	Multi-class	β	Split	Abdominal MRI				
				L. kid.	R. kid.	Spleen	Liver	Mean
ADNet	✗	-	1	81.97 ± 4.74	85.31 ± 3.05	76.99 ± 5.88	79.97 ± 4.01	81.06 ± 5.46
			2	78.80 ± 8.55	85.97 ± 3.14	36.94 ± 36.75	43.22 ± 37.75	61.23 ± 34.25
			3	83.91 ± 2.58	83.39 ± 3.53	79.56 ± 3.49	83.34 ± 4.68	82.55 ± 4.04
			4	76.09 ± 9.62	81.83 ± 3.16	76.98 ± 8.02	84.79 ± 3.05	79.92 ± 7.53
			5	77.07 ± 6.83	70.53 ± 17.8	69.69 ± 12.94	80.13 ± 6.33	74.35 ± 12.73
			Mean	**79.57 ± 7.55**	**81.41 ± 10.17**	**68.03 ± 24.05**	**74.29 ± 23.39**	**75.82 ± 5.20**
ADNet++	✓	1.0	1	81.24 ± 5.37	85.34 ± 3.00	78.19 ± 5.44	80.21 ± 3.94	81.25 ± 5.25
			2	79.35 ± 7.32	85.86 ± 3.08	38.44 ± 38.1	43.54 ± 37.78	61.80 ± 34.31
			3	84.08 ± 2.65	83.37 ± 3.50	80.23 ± 3.45	83.46 ± 4.71	82.78 ± 3.95
			4	79.83 ± 5.65	81.72 ± 3.23	77.53 ± 7.72	84.82 ± 3.13	80.97 ± 5.92
			5	78.32 ± 5.68	70.83 ± 17.62	69.84 ± 12.91	80.41 ± 6.21	74.85 ± 12.57
			Mean	**80.56 ± 5.89**	**81.42 ± 10.03**	**68.85 ± 24.24**	**74.49 ± 23.35**	**76.33 ± 5.08***
ADNet++	✓	0.3	1	85.26 ± 5.06	88.23 ± 1.70	86.48 ± 5.31	80.18 ± 5.02	85.04 ± 5.43
			2	86.17 ± 6.99	90.6 ± 1.93	42.92 ± 42.88	42.80 ± 39.38	65.62 ± 37.16
			3	89.66 ± 3.02	87.02 ± 1.82	87.97 ± 4.52	82.51 ± 5.15	86.79 ± 4.67
			4	90.71 ± 6.2	91.92 ± 3.11	83.46 ± 6.77	84.8 ± 3.61	87.72 ± 6.33
			5	82.91 ± 5.32	76.97 ± 20.07	77.63 ± 11.5	81.69 ± 4.89	79.8 ± 12.38
			Mean	**86.94 ± 6.19**	**86.95 ± 10.60**	**75.69 ± 26.35**	**74.40 ± 24.08**	**80.99 ± 5.97***
ADNet++	✓	$1 - \mathbf{U}(\mathbf{r})$	1	85.28 ± 5.02	89.29 ± 1.54	85.85 ± 4.23	80.35 ± 4.69	85.19 ± 5.20
			2	86.29 ± 6.98	90.61 ± 2.1	43.75 ± 43.75	43.78 ± 39.17	66.11 ± 37.11
			3	89.57 ± 2.03	87.43 ± 2.25	87.75 ± 4.14	82.68 ± 4.34	86.86 ± 4.22
			4	90.30 ± 4.22	88.97 ± 2.7	83.01 ± 6.81	84.93 ± 4.11	86.8 ± 5.55
			5	82.56 ± 6.64	76.79 ± 19.82	78.08 ± 10.56	82.51 ± 5.28	79.99 ± 12.28
			Mean	**86.80 ± 6.01**	**86.62 ± 10.37**	**75.69 ± 26.21**	**74.85 ± 23.82**	**80.99 ± 5.73***

Following (Kampffmeyer et al., 2016), to quantify the fidelity of the estimated uncertainty maps, we start by removing all voxels in the predictions and successively add voxels according to their estimated uncertainty, starting with the least uncertain voxels. Figure 5 shows how the segmentation performance decreases for all classes as more uncertain voxels are included[6]. This illustrates that voxels that are indicated by the uncertainty maps to be *certain* in fact are more probable of being correctly classified, whereas *uncertain* voxels have a higher probability of being falsely segmented. This means that the uncertainty maps can be used to quantify how much a prediction can be trusted.

6.4. Ablation study

In the following, we analyse the contribution of the different proposed components to the improved DSC, compared to the ADNet baseline. Tables 2 and 3 summarize the quantitative results for the CHAOS dataset and the MS-CMRSeg dataset, respectively.

[6]Note that the measure of the segmentation performance is accuracy (and not DSC) in this experiment. This because the denominator in Equation 13 varies as we include more and more voxels, making comparisons difficult.

Table 3: Quantitative evaluation of the proposed components' contribution on the MS-CMRSeg dataset. Mean DSC with standard deviations are reported for three runs per fold. * indicates that the increase in mean DSC, compared to the ADNet baseline, is statistically significant ($p < 0.05$).

Method	Multi-class	β	Split	Cardiac MRI			
				LV-BP	LV-MYO	RV	Mean
ADNet	✗	-	1	79.17 ± 5.82	51.61 ± 5.79	67.35 ± 8.9	66.04 ± 13.28
			2	81.04 ± 5.52	55.96 ± 4.86	66.15 ± 11.96	67.72 ± 13.11
			3	81.29 ± 5.72	55.79 ± 5.57	64.82 ± 7.43	67.3 ± 12.29
			4	80.57 ± 5.49	52.99 ± 5.85	66.10 ± 9.44	66.55 ± 13.35
			5	82.70 ± 4.19	52.05 ± 3.61	66.18 ± 12.0	66.98 ± 14.67
			Mean	**80.95 ± 5.50**	**53.68 ± 5.52**	**66.12 ± 10.14**	**66.92 ± 11.15**
ADNet++	✓	1.0	1	80.48 ± 7.0	55.27 ± 5.77	69.23 ± 8.45	68.33 ± 12.55
			2	81.40 ± 6.99	60.29 ± 5.81	67.54 ± 12.46	69.74 ± 12.49
			3	82.23 ± 5.14	60.17 ± 6.61	65.03 ± 10.24	69.14 ± 12.16
			4	80.02 ± 7.05	54.93 ± 7.21	67.21 ± 9.65	67.39 ± 13.03
			5	82.31 ± 5.32	56.56 ± 4.38	66.9 ± 11.81	68.59 ± 13.2
			Mean	**81.29 ± 6.43**	**57.44 ± 6.47**	**67.18 ± 10.71**	**68.64 ± 9.79***
ADNet++	✓	0.7	1	81.96 ± 7.72	57.80 ± 5.50	69.34 ± 9.28	69.7 ± 12.49
			2	82.67 ± 6.83	62.04 ± 4.80	68.1 ± 13.06	70.94 ± 12.45
			3	83.6 ± 4.97	61.82 ± 5.79	64.99 ± 9.50	70.13 ± 11.91
			4	81.86 ± 6.92	58.29 ± 6.08	67.09 ± 10.29	69.08 ± 12.57
			5	82.96 ± 5.76	58.37 ± 4.22	67.77 ± 12.66	69.7 ± 13.16
			Mean	**82.61 ± 6.55**	**59.66 ± 5.64**	**67.46 ± 11.17**	**69.91 ± 9.53***
ADNet++	✓	$1 - \mathbf{U}(\mathbf{r})$	1	82.01 ± 8.15	57.93 ± 5.51	69.45 ± 9.39	69.79 ± 12.58
			2	82.91 ± 6.02	62.19 ± 4.24	68.35 ± 13.44	71.15 ± 12.4
			3	83.46 ± 5.48	61.47 ± 6.67	64.46 ± 9.87	69.8 ± 12.33
			4	81.71 ± 6.67	58.94 ± 5.85	67.10 ± 10.34	69.25 ± 12.27
			5	82.77 ± 5.93	59.56 ± 4.5	67.83 ± 12.63	70.05 ± 12.8
			Mean	**82.57 ± 6.55**	**60.02 ± 5.66**	**67.44 ± 11.37**	**70.01 ± 9.38***

6.4.1. Binary vs. multi-class segmentation

In the first two rows of Table 2 and 3, we analyse the effect of moving from binary to multi-class segmentation. Any differences in segmentation results here are caused by the resolving of ambiguous voxel predictions, i.e. voxels previously assigned to multiple classes are now forced to choose one. In the CHAOS dataset, the issue of ambiguous voxels are most prominent between left kidney and spleen. This is because these organs share a boundary that often appears weak in the MRI scans. While the overall performance is only slightly improved when moving to the multi-class segmentation setting, the performance gains for left kidney and spleen are more visible (on average +0.99 and +0.82 percentage points, respectively). In the MS-CMRSeg dataset, all three classes share boundaries with one or both other organs. In the binary setting, the model typically over-segment all three classes, particularly hurting the performance of the LV-MYO because of its high surface-to-area ratio. When the model is forced to chose, the performance increases for all classes, especially for LV-MYO with +3.8 percentage points, yielding an average overall improvement of +1.7 percentage points. Figure 6 illustrates how the over-segmentation in the binary setting is improved in the multi-class setting for one example slice in the MS-CMRSeg dataset.

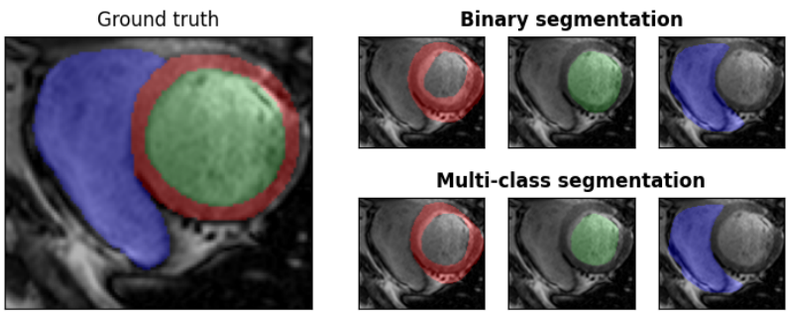


Figure 6: Qualitative evaluation of resolving ambiguous voxel predictions in a cropped example slice from the MS-CMR dataset. Where the model in the binary segmentation setting over-segments all three classes (red=LV-MYO, green=LV-BP, and blue=RV), it is in the multi-class setting forced to choose one class per voxel, resulting in less over-segmentation and higher DSC.

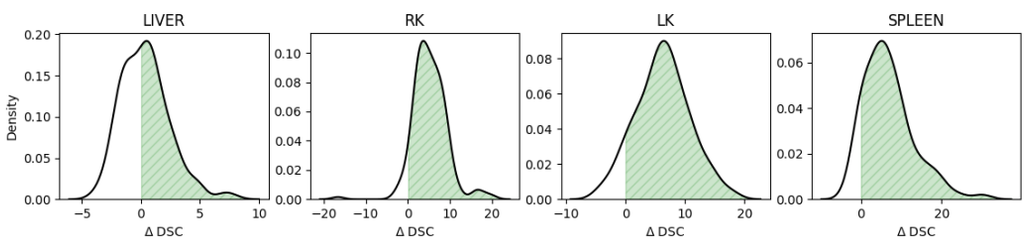


Figure 7: Distribution of Δ DSC for the segmentation results on the CHAOS dataset with and without feature refinement.

6.4.2. Feature refinement vs. no feature refinement

In rows three and four in Table 2 and Table 3, we investigate the effectiveness of the proposed feature refinement module on the two datasets. With a fixed β (dynamic $\beta(\boldsymbol{r})$), the module is able to improve the overall performance for both datasets, with +4.7 (+4.7) and +1.3 (+1.4) percentage points for the CHAOS dataset and the MS-CMRSeg dataset, respectively. Note that the fixed β is set to the optimal choice for the respective datasets, requiring a grid-search for parameter-tuning. Thus, while the improvement in overall segmentation performance is similar, the dynamic $\beta(\boldsymbol{r}) = 1 - \mathbf{U}(\boldsymbol{r})$ has the important advantage that it is computed automatically and does not need further fine-tuning.

Figure 7 shows the distribution of the difference in DSC (Δ DSC) for predictions *with* and *without* feature refinement (with a dynamic $\beta(\boldsymbol{r})$), for each class in the CHAOS dataset. For most cases, the feature refinement improves the DSC (green regions). However, the effect is split for the liver class, resulting in no overall improvement. This is related to the difficulty in capturing the liver (especially its left lobe) with supervoxels.

In the following section, we investigate the choice of β and how it effects the final segmentation results for different supervoxel settings.

Choice of β. The choice of β controls the extent of the feature refinement, from no refinement at $\beta = 1.0$ to moving the features all the way to their corresponding supervoxel center at $\beta = 0.0$. Figure 8 shows the prediction results for one example slice in the CHAOS dataset as we adjust the value of a fixed β from 0.0 to 1.0. For $\beta = 1.0$, we see that the model has difficulty with

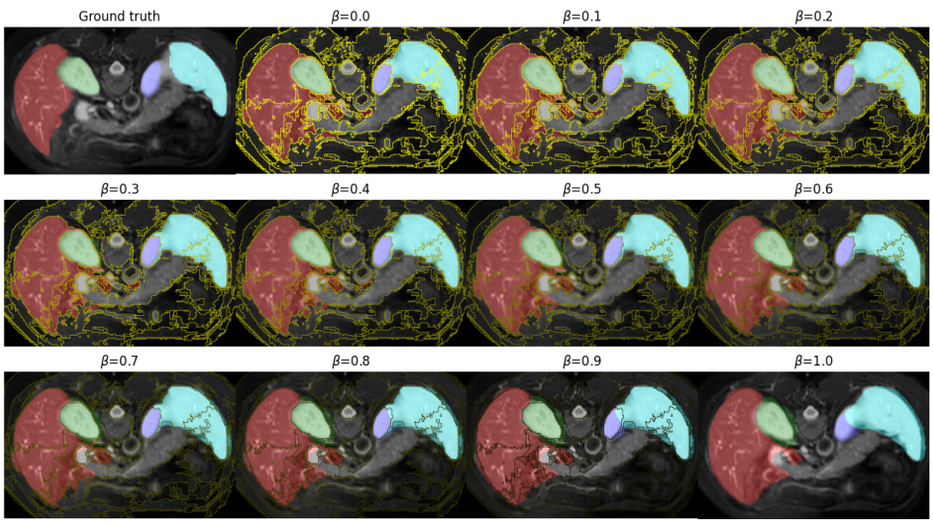


Figure 8: Qualitative evaluation of the feature refinement with β as a fixed constant for all voxels. The example slice is taken from the CHAOS dataset and is overlaid by the corresponding supervoxel boundaries (yellow) and the resulting segmentation masks (red=liver, green=right kidney, purple=left kidney, and blue=spleen).

locating the exact class boundaries, even when the edges in the input image are strong (e.g. the boundaries between right kidney and the background). As we reduce β, we see that the segmentation boundaries become gradually sharper. However, as β approaches 0.0, the prediction relies completely on the supervoxel segmentation, which might be faulty, especially in regions where boundaries in the input image are weak.

As discussed in Section 4.3, a dynamic $\beta(\mathbf{r}) = 1 - \mathbf{U}(\mathbf{r})$ has the potential to increase the flexibility of the feature refinement by allowing different voxels to move with different step lengths, depending on the model's uncertainty: For voxels in regions where the model is unsure about its initial prediction, we will pay more attention to the region information in the input space.

To systematically examine the effect of the choice β (fixed and dynamic) for different supervoxel sizes, we perform a grid search. Figure 9 and 10 show the results for the CHAOS dataset and MS-CMRSeg dataset, respectively. The top row in both figures display the grid-search over supervoxel size and a range of *fixed* betas, $\beta \in [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$, while the bottom rows display a line-search over supervoxel size with a *dynamic* beta computed via Equation 12.

For the CHAOS dataset, in particular, we see that the optimal combination of supervoxel size and fixed beta varies a lot between the classes (top row, Figure 9). For instance, the liver class prefers smaller supervoxels and a high β, whereas the spleen class prefers somewhat larger supervoxels and a lower β. This can be connected to the typical supervoxel quality for these organs: Weak edges in the liver result in unreliable supervoxels, making it "safer" to go with small supervoxels and rely more on the original representation. The spleen, on the other hand, is easier captured by the supervoxels and the confusion between left kidney and spleen in the feature space can be resolved by relying more on the supervoxels.

The line-searches over supervoxel size with a dynamic β (bottom rows in Figure 9 and 10) show that with a dynamic β, the results across different supervoxel sizes are more stable for both datasets. They further illustrate that exploiting the uncertainty map is an efficient approach to

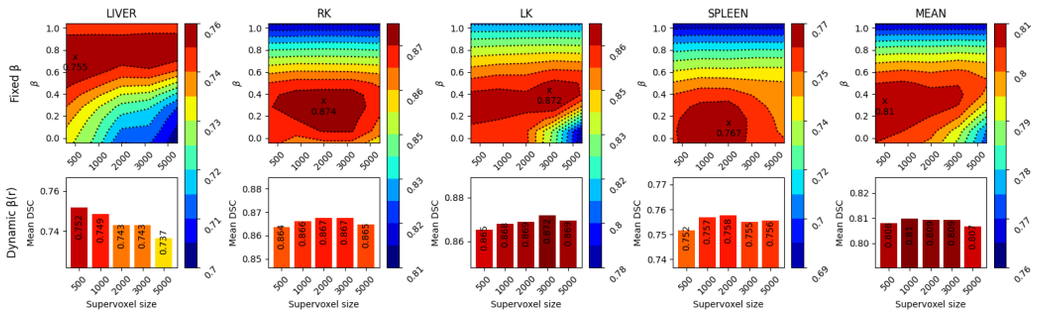


Figure 9: Parameter sensitivity of feature-refinement module on the *CHAOS dataset*. Top: Grid-search over supervoxel sizes and a range of *fixed* betas. Bottom: Line-search over supervoxel sizes with a *dynamic* beta automatically computed from uncertainty maps.

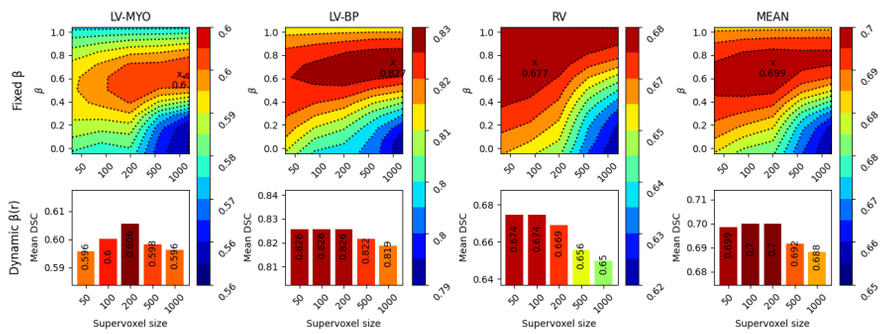


Figure 10: Parameter sensitivity of feature-refinement module on the *MS-CMRSeg dataset*. Top: Grid-search over supervoxel sizes and a range of *fixed* betas. Bottom: Line-search over supervoxel sizes with a *dynamic* beta automatically computed from uncertainty maps.

automatically decide $\beta(\boldsymbol{r})$.

Figure 11 shows the distribution of $\beta(\mathbf{r})$ for each class c in the CHAOS dataset, illustrating how the features of the different organs are refined with a greater or lesser influence of the supervoxel information. For instance, most of the voxels belonging to right kidney get a high value of beta, meaning that they are experiencing a lower degree of feature refinement. This is because the prediction of the right kidney class typically is quite certain, with the exception of the edge voxels, which contribute to the long tail of the distribution in Figure 11.

7. Conclusion and outlook

Prototypical few-shot learning is an emerging research direction within medical image segmentation that offers promising results without requiring large labeled datasets. In this work, we identify three weaknesses of current prototypical FSS models for medical image segmentation and propose new methodology to overcome these. Specifically, we propose the ADNet++, the first model that performs one-step multi-class segmentation and that provides uncertainty

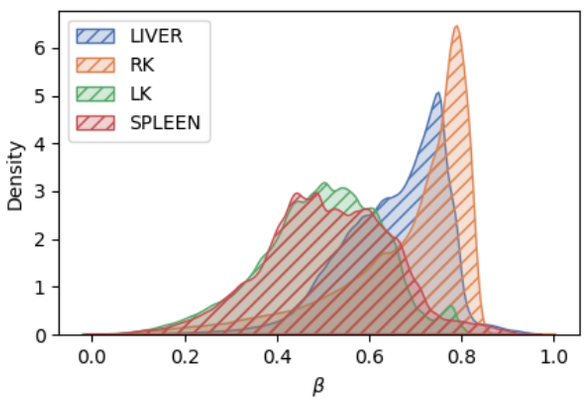


Figure 11: Distribution of β for the different classes in the CHAOS dataset, when decided automatically from the uncertainty maps: $\beta(\mathbf{r}) = 1 - \mathbf{U}(\mathbf{r})$.

maps to accompany its predictions. In addition to indicate the model's confidence in the predictions, thereby increasing the models trustworthiness, the uncertainty maps are further exploited to guide the proposed feature refinement that leverages structural information in the input space to provide more accurate segmentation results. The proposed model significantly improves the current state-of-the-art 3D FSS model for the tasks of MRI-based abdominal organ segmentation and cardiac segmentation.

In future work, it would be interesting to explore methods that can make the feature-refinement module more robust to supervoxel quality, as its success largely depends on it. Instead of relying on *one* set of supervoxels, a potential approach could be to explore *multi-scale* supervoxels, e.g. supervoxels of different sizes. Furthermore, given the model-agnostic nature of our proposed modules, future work should implement and evaluate their fidelity in other FSS frameworks.

Acknowledgements

This work was supported by The Research Council of Norway (RCN), through its Centre for Research-based Innovation funding scheme [grant number 309439] and Consortium Partners; RCN FRIPRO [grant number 315029]; RCN IKTPLUSS [grant number 303514]; and the UiT Thematic Initiative.

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. Slic superpixels compared to state-of-the-art superpixel methods. IEEE transactions on pattern analysis and machine intelligence 34, 2274–2282.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient graph-based image segmentation. International journal of computer vision 59, 167–181.

Gal, Y., 2016. Uncertainty in deep learning. Ph.D. thesis. University of Cambridge.

Gal, Y., Ghahramani, Z., 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: international conference on machine learning, PMLR. pp. 1050–1059.

Gonzalez, Y., Shen, C., Jung, H., Nguyen, D., Jiang, S.B., Albuquerque, K., Jia, X., 2021. Semi-automatic sigmoid colon segmentation in ct for radiation therapy treatment planning via an iterative 2.5-d deep learning approach. Medical Image Analysis 68, 101896.

Hansen, S., Gautam, S., Jenssen, R., Kampffmeyer, M., 2022. Anomaly detection-inspired few-shot medical image segmentation through self-supervision with supervoxels. Medical Image Analysis 78, 102385.

Harper, R., Southern, J., 2020. A bayesian deep learning framework for end-to-end prediction of emotion from heartbeat. IEEE transactions on affective computing .

van Hespen, K.M., Zwanenburg, J.J., Hendrikse, J., Kuijf, H.J., 2021. Subvoxel vessel wall thickness measurements of the intracranial arteries using a convolutional neural network. Medical Image Analysis 67, 101818.

Johnander, J., Edstedt, J., Danelljan, M., Felsberg, M., Khan, F.S., 2021. Deep gaussian processes for few-shot segmentation. arXiv preprint arXiv:2103.16549 .

Kampffmeyer, M., Salberg, A.B., Jenssen, R., 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 1–9.

Karimi, D., Zeng, Q., Mathur, P., Avinash, A., Mahdavi, S., Spadinger, I., Abolmaesumi, P., Salcudean, S.E., 2019. Accurate and robust deep learning-based segmentation of the prostate clinical target volume in ultrasound images. Medical image analysis 57, 186–196.

Kavur, A.E., Gezer, N.S., Barış, M., Aslan, S., Conze, P.H., Groza, V., Pham, D.D., Chatterjee, S., Ernst, P., Özkan, S., Baydar, B., Lachinov, D., Han, S., Pauli, J., Isensee, F., Perkonigg, M., Sathish, R., Rajan, R., Sheet, D., Dovletov, G., Speck, O., Nürnberger, A., Maier-Hein, K.H., Bozdağı Akar, G., Ünal, G., Dicle, O., Selver, M.A., 2021. CHAOS Challenge - combined (CT-MR) healthy abdominal organ segmentation. Medical Image Analysis 69, 101950. URL: http://www.sciencedirect.com/science/article/pii/S1361841520303145, doi:https://doi.org/10.1016/j.media.2020.101950.

Kavur, A.E., Gezer, N.S., Barış, M., Şahin, Y., Özkan, S., Baydar, B., Yüksel, U., Kılıkçıer, , Olut, , Bozdağı Akar, G., Ünal, G., Dicle, O., Selver, M.A., 2020. Comparison of semi-automatic and deep learning based automatic methods for liver segmentation in living liver transplant donors. Diagnostic and Interventional Radiology 26, 11–21. URL: https://doi.org/10.5152/dir.2019.19025, doi:10.5152/dir.2019.19.

Kavur, A.E., Selver, M.A., Dicle, O., Barış, M., Gezer, N.S., 2019. CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation Challenge Data. URL: https://doi.org/10.5281/zenodo.3362844, doi:10.5281/zenodo.3362844.

Kim, S., Chikontwe, P., Park, S.H., 2021. Uncertainty-aware semi-supervised few shot segmentation. arXiv preprint arXiv:2110.08954 .

Kompa, B., Snoek, J., Beam, A.L., 2021. Second opinion needed: communicating uncertainty in medical machine learning. NPJ Digital Medicine 4, 1–6.

Lakshminarayanan, B., Pritzel, A., Blundell, C., 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. Advances in neural information processing systems 30.

Militello, C., Rundo, L., Toia, P., Conti, V., Russo, G., Filorizzo, C., Maffei, E., Cademartiri, F., La Grutta, L., Midiri, M., et al., 2019. A semi-automatic approach for epicardial adipose tissue segmentation and quantification on cardiac ct scans. Computers in biology and medicine 114, 103424.

Ouyang, C., Biffi, C., Chen, C., Kart, T., Qiu, H., Rueckert, D., 2022. Self-supervised learning for few-shot medical image segmentation. IEEE Transactions on Medical Imaging .

Roy, A.G., Siddiqui, S., Pölsterl, S., Navab, N., Wachinger, C., 2020. "squeeze & excite" guided few-shot segmentation of volumetric images. Medical image analysis 59, 101587.

Shen, C., Nguyen, D., Zhou, Z., Jiang, S.B., Dong, B., Jia, X., 2020. An introduction to deep learning in medical physics: advantages, potential, and challenges. Physics in Medicine & Biology 65, 05TR01.

Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning, in: Advances in neural information processing systems, pp. 4077–4087.

Tang, H., Liu, X., Sun, S., Yan, X., Xie, X., 2021. Recurrent mask refinement for few-shot medical image segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3918–3928.

Tsochatzidis, L., Koutla, P., Costaridou, L., Pratikakis, I., 2021. Integrating segmentation information into cnn for breast cancer diagnosis of mammographic masses. Computer methods and programs in biomedicine 200, 105913.

Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J., 2019. Panet: Few-shot image semantic segmentation with prototype alignment, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9197–9206.

Wickstrøm, K., Kampffmeyer, M., Jenssen, R., 2020. Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps. Medical image analysis 60, 101619.

Wilcoxon, F., 1992. Individual comparisons by ranking methods, in: Breakthroughs in statistics. Springer, pp. 196–202.

Yu, Q., Dang, K., Tajbakhsh, N., Terzopoulos, D., Ding, X., 2021. A location-sensitive local prototype network for few-shot medical image segmentation, in: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), IEEE. pp. 262–266.

Zhuang, X., 2016. Multivariate mixture model for cardiac segmentation from multi-sequence mri, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 581–588.
Zhuang, X., 2018. Multivariate mixture model for myocardial segmentation combining multi-source images. IEEE transactions on pattern analysis and machine intelligence 41, 2933–2946.

Bibliography

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282.

Afrasiyabi, A., Larochelle, H., Lalonde, J.-F., and Gagné, C. (2022). Matching feature sets for few-shot image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9014–9024.

Ahn, E., Feng, D., and Kim, J. (2021). A spatial guided self-supervised clustering network for medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 379–388. Springer.

Bagci, U., Udupa, J. K., Mendhiratta, N., Foster, B., Xu, Z., Yao, J., Chen, X., and Mollura, D. J. (2013). Joint segmentation of anatomical and functional images: Applications in quantification of lesions from pet, pet-ct, mri-pet, and mri-pet-ct images. *Medical image analysis*, 17(8):929–945.

Bengio, Y., Vincent, P., Paiement, J.-F., Delalleau, O., Ouimet, M., and Le Roux, N. (2003). *Spectral clustering and kernel PCA are learning eigenfunctions*, volume 1239. CIRANO.

Bercovich, E. and Javitt, M. C. (2018). Medical imaging: from roentgen to the digital revolution, and beyond. *Rambam Maimonides medical journal*, 9(4).

Beyer, T., Freudenberg, L. S., Czernin, J., and Townsend, D. W. (2011). The future of hybrid imaging—part 3: Pet/mr, small-animal imaging and beyond. *Insights into imaging*, 2(3):235–246.

Boudiaf, M., Kervadec, H., Masud, Z. I., Piantanida, P., Ben Ayed, I., and Dolz, J. (2021). Few-shot segmentation without meta-learning: A good transductive inference is all you need? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13979–13988.

Box, G. E. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, 26(2):211–243.

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017a). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.

Chen, M., Yan, Q., and Qin, M. (2017b). A segmentation of brain mri images utilizing intensity and contextual information by markov random field. *Computer Assisted Surgery*, 22(sup1):200–211.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Chen, W.-Y., Liu, Y.-C., Kira, Z., Wang, Y.-C. F., and Huang, J.-B. (2019). A closer look at few-shot classification. In *International Conference on Learning Representations*.

Chen, X. and He, K. (2021). Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758.

Chen, X., Sun, S., Bai, N., Han, K., Liu, Q., Yao, S., Tang, H., Zhang, C., Lu, Z., Huang, Q., et al. (2021). A deep learning-based auto-segmentation system for organs-at-risk on whole-body computed tomography images for radiation therapy. *Radiotherapy and Oncology*, 160:175–184.

Cho, J. H., Mall, U., Bala, K., and Hariharan, B. (2021). Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16794–16804.

Chu, J., Min, H., Liu, L., and Lu, W. (2015). A novel computer aided breast mass detection scheme based on morphological enhancement and slic superpixel segmentation. *Medical physics*, 42(7):3859–3869.

Crișan, G., Moldovean-Cioroianu, N. S., Timaru, D.-G., Andrieș, G., Căinap, C., and Chiș, V. (2022). Radiopharmaceuticals for pet and spect imaging: A literature review over the last decade. *International Journal of Molecular Sciences*, 23(9):5023.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and*

pattern recognition (CVPR'05), volume 1, pages 886–893. Ieee.

Daniels, C. J. and Gallagher, F. A. (2017). Unsupervised segmentation of 5d hyperpolarized carbon-13 mri data using a fuzzy markov random field model. *IEEE Transactions on Medical Imaging*, 37(4):840–850.

Day, E., Betler, J., Parda, D., Reitz, B., Kirichenko, A., Mohammadi, S., and Miften, M. (2009). A region growing method for tumor volume segmentation on pet images for rectal and anal cancer patients. *Medical physics*, 36(10):4349–4358.

Dehmeshki, J., Amin, H., Valdivieso, M., and Ye, X. (2008). Segmentation of pulmonary nodules in thoracic ct scans: a region growing approach. *IEEE transactions on medical imaging*, 27(4):467–480.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Dhillon, G. S., Chaudhari, P., Ravichandran, A., and Soatto, S. (2020). A baseline for few-shot image classification. In *International Conference on Learning Representations*.

Dietterich, T. (2003). Machine learning in nature encyclopedia of cognitive science.

Doersch, C., Gupta, A., and Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430.

Doersch, C., Gupta, A., and Zisserman, A. (2020). Crosstransformers: spatially-aware few-shot transfer. *Advances in Neural Information Processing Systems*, 33:21981–21993.

Dong, N., Kampffmeyer, M., and Voiculescu, I. (2021). Self-supervised multi-task representation learning for sequential medical images. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 779–794. Springer.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Edwards, H. and Storkey, A. (2016). Towards a neural statistician. *arXiv*

preprint arXiv:1606.02185.

Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., Liu, Y., Topol, E., Dean, J., and Socher, R. (2021). Deep learning-enabled medical computer vision. *NPJ digital medicine*, 4(1):1–9.

Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181.

Feng, Y., Zhao, H., Li, X., Zhang, X., and Li, H. (2017). A multi-scale 3d otsu thresholding algorithm for medical image segmentation. *Digital Signal Processing*, 60:186–199.

Fu, Y., Lei, Y., Wang, T., Curran, W. J., Liu, T., and Yang, X. (2021). A review of deep learning based methods for medical image multi-organ segmentation. *Physica Medica*, 85:107–122.

Gal, Y. (2016). *Uncertainty in deep learning*. PhD thesis, University of Cambridge.

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

Gautam, S., Boubekki, A., Hansen, S., Salahuddin, S. A., Jenssen, R., Höhne, M. M., and Kampffmeyer, M. (2022a). Protovae: A trustworthy self-explainable prototypical variational model. *Advances in Neural Information Processing Systems*, 36.

Gautam, S., Höhne, M. M.-C., Hansen, S., Jenssen, R., and Kampffmeyer, M. (2022b). Demonstrating the risk of imbalanced datasets in chest x-ray image-based diagnostics by prototypical relevance propagation. In *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE.

Gawlikowski, J., Tassi, C. R. N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., et al. (2021). A survey of uncertainty in deep neural networks. *arXiv preprint arXiv:2107.03342*.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings.

Gonzalez, R. and Woods, R. (2008). *Digital Image Processing*. Prentice Hall.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Greenwald, N. F., Miller, G., Moen, E., Kong, A., Kagel, A., Dougherty, T., Fullaway, C. C., McIntosh, B. J., Leow, K. X., Schwartz, M. S., et al. (2022). Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature biotechnology*, 40(4):555–565.

Gui, L., Bardes, A., Salakhutdinov, R., Hauptmann, A., Hebert, M., and Wang, Y.-X. (2021). Learning to hallucinate examples from extrinsic and intrinsic supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8701–8711.

Hara, K., Kataoka, H., and Satoh, Y. (2018). Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555.

Hariharan, B. and Girshick, R. (2017). Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE international conference on computer vision*, pages 3018–3027.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.-R., and Samek, W. (2022). xxai-beyond explainable artificial intelligence. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pages 3–10. Springer.

Iglesias, J. E., Sabuncu, M. R., and Van Leemput, K. (2013). A unified framework for cross-modality multi-atlas segmentation of brain mri. *Medical image*

analysis, 17(8):1181–1191.

Iniewski, K. (2009). *Medical imaging: principles, detectors, and electronics*. John Wiley & Sons.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR.

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666.

Judenhofer, M. S., Wehrl, H. F., Newport, D. F., Catana, C., Siegel, S. B., Becker, M., Thielscher, A., Kneilling, M., Lichy, M. P., Eichner, M., et al. (2008). Simultaneous pet-mri: a new approach for functional and morphological imaging. *Nature medicine*, 14(4):459–465.

Kang, D. and Cho, M. (2022). Integrative few-shot learning for classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9979–9990.

Kang, D., Kwon, H., Min, J., and Cho, M. (2021). Relational embedding for few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8822–8833.

Kavur, A. E., Gezer, N. S., Barış, M., Aslan, S., Conze, P.-H., Groza, V., Pham, D. D., Chatterjee, S., Ernst, P., Özkan, S., Baydar, B., Lachinov, D., Han, S., Pauli, J., Isensee, F., Perkonigg, M., Sathish, R., Rajan, R., Sheet, D., Dovletov, G., Speck, O., Nürnberger, A., Maier-Hein, K. H., Bozdağı Akar, G., Ünal, G., Dicle, O., and Selver, M. A. (2021). CHAOS Challenge - combined (CT-MR) healthy abdominal organ segmentation. *Medical Image Analysis*, 69:101950.

Kavur, A. E., Gezer, N. S., Barış, M., Şahin, Y., Özkan, S., Baydar, B., Yüksel, U., Kılıkçıer, , Olut, , Bozdağı Akar, G., Ünal, G., Dicle, O., and Selver, M. A. (2020). Comparison of semi-automatic and deep learning based automatic methods for liver segmentation in living liver transplant donors. *Diagnostic and Interventional Radiology*, 26:11–21.

Kavur, A. E., Selver, M. A., Dicle, O., Barış, M., and Gezer, N. S. (2019). CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation Challenge Data.

Kazemzadeh, S., Yu, J., Jamshy, S., Pilgrim, R., Nabulsi, Z., Chen, C., Beladia, N., Lau, C., McKinney, S. M., Hughes, T., et al. (2022). Deep learning detection

of active pulmonary tuberculosis at chest radiography matched the clinical performance of radiologists. *Radiology*, page 212213.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Komodakis, N. and Gidaris, S. (2018). Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations (ICLR)*.

Kumar, S. N., Fred, A. L., and Varghese, P. S. (2019). Suspicious lesion segmentation on brain, mammograms and breast mr images using new optimized spatial feature based super-pixel fuzzy c-means clustering. *Journal of digital imaging*, 32(2):322–335.

Kuttner, S., Lassen, M. L., Øen, S. K., Sundset, R., Beyer, T., and Eikenes, L. (2020). Quantitative pet/mr imaging of lung cancer in the presence of artifacts in the mr-based attenuation correction maps. *Acta Radiologica*, 61(1):11–20.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Leibfarth, S., Eckert, F., Welz, S., Siegel, C., Schmidt, H., Schwenzer, N., Zips, D., and Thorwarth, D. (2015). Automatic delineation of tumor volumes by co-segmentation of combined pet/mr data. *Physics in Medicine & Biology*, 60(14):5399.

Li, G., Jampani, V., Sevilla-Lara, L., Sun, D., Kim, J., and Kim, J. (2021). Adaptive prototype learning and allocation for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8334–8343.

Liu, Y., Zhang, X., Zhang, S., and He, X. (2020). Part-aware prototype network for few-shot semantic segmentation. In *European Conference on Computer Vision*, pages 142–158. Springer.

Lötjönen, J. M., Wolz, R., Koikkalainen, J. R., Thurfjell, L., Waldemar, G., Soininen, H., Rueckert, D., Initiative, A. D. N., et al. (2010). Fast and robust multi-atlas segmentation of brain magnetic resonance images. *Neuroimage*, 49(3):2352–2365.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. IEEE.

Makropoulos, A., Gousias, I. S., Ledig, C., Aljabar, P., Serag, A., Hajnal, J. V., Edwards, A. D., Counsell, S. J., and Rueckert, D. (2014). Automatic whole brain mri segmentation of the developing neonatal brain. *IEEE transactions on medical imaging*, 33(9):1818–1831.

McDonald, R. J., Schwartz, K. M., Eckel, L. J., Diehn, F. E., Hunt, C. H., Bartholmai, B. J., Erickson, B. J., and Kallmes, D. F. (2015). The effects of changes in utilization and technological advancements of cross-sectional imaging on radiologist workload. *Academic radiology*, 22(9):1191–1198.

Mercieca, S., Belderbos, J., Van Loon, J., Gilhuijs, K., Julyan, P., and Van Herk, M. (2018). Comparison of suvmax and suvpeak based segmentation to determine primary lung tumour volume on fdg pet-ct correlated with pathology data. *Radiotherapy and Oncology*, 129(2):227–233.

Min, J., Kang, D., and Cho, M. (2021). Hypercorrelation squeeze for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6941–6952.

Misra, I. and Maaten, L. v. d. (2020). Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill Professional, New York, NY.

Morais, P., Vilaça, J. L., Queirós, S., Bourier, F., Deisenhofer, I., Tavares, J. M. R., and D'hooge, J. (2017). A competitive strategy for atrial and aortic tract segmentation based on deformable models. *Medical image analysis*, 42:102–116.

Neal, R. M. (2012). *Bayesian learning for neural networks*. PhD thesis, University of Toronto.

Nelms, B. E., Tomé, W. A., Robinson, G., and Wheeler, J. (2012). Variations in the contouring of organs at risk: test case from a patient with oropharyngeal cancer. *International Journal of Radiation Oncology* Biology* Physics*, 82(1):368–378.

Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer.

Ouyang, C., Biffi, C., Chen, C., Kart, T., Qiu, H., and Rueckert, D. (2022). Self-

supervised learning for few-shot medical image segmentation. *IEEE Transactions on Medical Imaging*.

Parnami, A. and Lee, M. (2022). Learning from few examples: A summary of approaches to few-shot learning. *arXiv preprint arXiv:2203.04291*.

Peng, J. and Wang, Y. (2021). Medical image segmentation with limited supervision: a review of deep network models. *IEEE Access*, 9:36827–36851.

Pham, D. L., Xu, C., and Prince, J. L. (2000). Current methods in medical image segmentation. *Rev. Biomed. Eng*.

Rahmati, P., Adler, A., and Hamarneh, G. (2012). Mammography segmentation with maximum likelihood active contours. *Medical image analysis*, 16(6):1167–1186.

Rebouças Filho, P. P., Cortez, P. C., da Silva Barros, A. C., Albuquerque, V. H. C., and Tavares, J. M. R. (2017). Novel and powerful 3d adaptive crisp active contour method applied in the segmentation of ct lung images. *Medical image analysis*, 35:503–516.

Rehman, Z. U., Naqvi, S. S., Khan, T. M., Khan, M. A., and Bashir, T. (2019). Fully automated multi-parametric brain tumour segmentation using superpixel based classification. *Expert systems with applications*, 118:598–613.

Rezende, D., Danihelka, I., Gregor, K., Wierstra, D., et al. (2016). One-shot generalization in deep generative models. In *International conference on machine learning*, pages 1521–1529. PMLR.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Salzer, R. (2012). *Biomedical imaging: principles and applications*. John Wiley & Sons.

Sbei, A., ElBedoui, K., Barhoumi, W., Maksud, P., and Maktouf, C. (2017). Hybrid pet/mri co-segmentation based on joint fuzzy connectedness and graph cut. *Computer Methods and Programs in Biomedicine*, 149:29–41.

Sbei, A., ElBedoui, K., Barhoumi, W., and Maktouf, C. (2020). Gradient-based generation of intermediate images for heterogeneous tumor segmentation within hybrid pet/mri scans. *Computers in Biology and Medicine*, 119:103669.

Schick, F. (2022). Automatic segmentation and volumetric assessment of internal organs and fatty tissue: what are the benefits? *Magnetic Resonance Materials in Physics, Biology and Medicine*, 35(2):187–192.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Siemens Healthineers (2021). Features, data, and algorithms ai-rad companion chest ct va13. White paper, Siemens Healthineers.

Simpson, D. L., Bui-Mansfield, L. T., and Bank, K. P. (2017). Fdg pet/ct: artifacts and pitfalls. *Contemporary Diagnostic Radiology*, 40(5):1–7.

Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Subudhi, S., Patro, R. N., Biswal, P. K., and Dell'Acqua, F. (2021). A survey on superpixel segmentation as a preprocessing step in hyperspectral image analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:5015–5035.

Sun, C., Lee, J. S., and Zhang, M. (2008). Magnetic nanoparticles in mr imaging and drug delivery. *Advanced drug delivery reviews*, 60(11):1252–1265.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. (2018). Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208.

Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., and Bray, F. (2021). Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 71(3):209–249.

Tang, H., Liu, X., Sun, S., Yan, X., and Xie, X. (2021). Recurrent mask refinement for few-shot medical image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3918–3928.

Tian, Z., Zhao, H., Shu, M., Yang, Z., Li, R., and Jia, J. (2020). Prior guided feature enrichment network for few-shot segmentation. *IEEE transactions on pattern analysis and machine intelligence*.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

Tong, N., Lu, H., Zhang, L., and Ruan, X. (2014). Saliency detection with multi-scale superpixels. *IEEE Signal Processing Letters*, 21(9):1035–1039.

Vanschoren, J. (2018). Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*.

Vedaldi, A. and Soatto, S. (2008). Quick shift and kernel methods for mode seeking. In *European conference on computer vision*, pages 705–718. Springer.

Velazquez, E. R., Parmar, C., Jermoumi, M., Mak, R. H., van Baardwijk, A., Fennessy, F. M., Lewis, J. H., De Ruysscher, D., Kikinis, R., Lambin, P., et al. (2013). Volumetric ct-based segmentation of nsclc using 3d-slicer. *Scientific reports*, 3(1):1–7.

Vincent, L. and Soille, P. (1991). Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(06):583–598.

Vondrick, C., Khosla, A., Malisiewicz, T., and Torralba, A. (2013). Hoggles: Visualizing object detection features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8.

Wallyn, J., Anton, N., Akram, S., and Vandamme, T. F. (2019). Biomedical imaging: principles, technologies, clinical aspects, contrast agents, limitations and future trends in nanomedicines. *Pharmaceutical Research*, 36(6):1–31.

Wang, H., Zhang, X., Hu, Y., Yang, Y., Cao, X., and Zhen, X. (2020). Few-shot semantic segmentation with democratic attention networks. In *European Conference on Computer Vision*, pages 730–746. Springer.

Wang, K., Liew, J. H., Zou, Y., Zhou, D., and Feng, J. (2019). Panet: Few-shot image semantic segmentation with prototype alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Wang, S., Li, C., Wang, R., Liu, Z., Wang, M., Tan, H., Wu, Y., Liu, X., Sun, H., Yang, R., et al. (2021a). Annotation-efficient deep learning for automatic medical image segmentation. *Nature communications*, 12(1):1–13.

Wang, X., Zhang, R., Shen, C., Kong, T., and Li, L. (2021b). Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033.

Wang, Z., Li, Q., Zhang, G., Wan, P., Zheng, W., Wang, N., Gong, M., and Liu, T. (2022). Exploring set similarity for dense self-supervised representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16590–16599.

Wickstrøm, K., Kampffmeyer, M., and Jenssen, R. (2020). Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps. *Medical image analysis*, 60:101619.

Wu, J., Poehlman, S., Noseworthy, M. D., and Kamath, M. V. (2008). Texture feature based automated seeded region growing in abdominal mri segmentation. In *2008 International Conference on BioMedical Engineering and Informatics*, volume 2, pages 263–267. IEEE.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500.

Xu, Z., Bagci, U., Udupa, J. K., and Mollura, D. J. (2015). Fuzzy connectedness image co-segmentation for hybridpet/mri and pet/ct scans. In *Computational Methods for Molecular Imaging*, pages 15–24. Springer.

Yang, B., Liu, C., Li, B., Jiao, J., and Ye, Q. (2020). Prototype mixture models for few-shot semantic segmentation. In *European Conference on Computer Vision*, pages 763–778. Springer.

Ye, X., Beddoe, G., and Slabaugh, G. (2010). Automatic graph cut segmentation of lesions in ct using mean shift superpixels. *International journal of biomedical imaging*, 2010.

Yu, Q., Dang, K., Tajbakhsh, N., Terzopoulos, D., and Ding, X. (2021). A location-sensitive local prototype network for few-shot medical image segmentation. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 262–266. IEEE.

Zhang, B., Xiao, J., and Qin, T. (2021a). Self-guided and cross-guided learning for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8312–8321.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2021b). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.

Zhang, C., Lin, G., Liu, F., Guo, J., Wu, Q., and Yao, R. (2019a). Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9587–9595.

Zhang, C., Lin, G., Liu, F., Yao, R., and Shen, C. (2019b). Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5217–5226.

Zhang, G., Kang, G., Yang, Y., and Wei, Y. (2021c). Few-shot segmentation via cycle-consistent transformer. *Advances in Neural Information Processing Systems*, 34:21984–21996.

Zhao, Z., Zhao, J., Song, K., Hussain, A., Du, Q., Dong, Y., Liu, J., and Yang, X. (2020). Joint dbn and fuzzy c-means unsupervised deep clustering for lung cancer patient stratification. *Engineering Applications of Artificial Intelligence*, 91:103571.

Zhu, S., Gilbert, M., Chetty, I., and Siddiqui, F. (2022). The 2021 landscape of fda-approved artificial intelligence/machine learning-enabled medical devices: An analysis of the characteristics and intended use. *International Journal of Medical Informatics*, 165:104828.

Ziko, I., Dolz, J., Granger, E., and Ayed, I. B. (2020). Laplacian regularized few-shot learning. In *International conference on machine learning*, pages 11660–11670. PMLR.