UNIVERSITY OF DUISBURG-ESSEN

DISSERTATION

Anatomical Priors for Fully Automated Medical Image Segmentation

---

Von der Fakultät für Ingenieurwissenschaften,
Abteilung Informatik und Angewandte Kognitionswissenschaft
der Universität Duisburg-Essen

zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von

Duc Duy Pham
aus
Hamm

1. Gutachter: Prof. Dr. Josef PAULI
2. Gutachter: Prof. Dr. Xiaoyi JIANG

Tag der mündlichen Prüfung: 13. Oktober 2022

In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of University of Duisburg-Essen's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to `http://www.ieee.org/publications_standards/publications/rights/rights_link.html` to learn how to obtain a License from RightsLink. If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

"We can only see a short distance ahead, but we can see plenty there that needs to be done."

Alan Turing

Acknowledgments

First and foremost, I would like to express my gratitude towards my supervisor, Prof. Dr. Josef Pauli, for the opportunity to pursue this research path towards a doctoral degree and for his trust in me that I could succeed. I am also truly grateful for the degree of freedom I was entrusted with, both in research and teaching. I have always appreciated his calm and kind personality, with which he offered encouragement and valuable insights on this journey. His expertise and guidance considerably contributed to the success of this dissertation.

I would also like to thank Prof. Dr. Xiaoyi Jiang for his interest in my work and agreeing to take part in the review and oral examination process.
Moreover, I am thankful towards Prof. Dr. Marcus Jäger, Prof. Dr. Stefan Landgraeber, and Dr. Sebastian Serong for their suggestions of clinically relevant research topics and for the acquisition and provision of medical image data for my research. I also appreciate the collaboration with all further colleagues, whom I had the pleasure to work with.

I had a great time at the Intelligent Systems Group and I had the joy of working together with amazing people over these years. Very special thanks go to Gurbandurdy Dovletov, Stefan Lörcks, and Tobias Hegemann for the joyful time we had working together. I dearly miss the office shenanigans and sometimes even the Life Kinetik sessions after our lunch break. On the other hand, I am also very grateful for all the fruitful (and also not so fruitful) discussions we had. Of course this also goes for Martin Moder, who always comes up with very exciting scientific topics, and Marius Bock, who is never short of an unbelievable story to share.
Thanks to Fatih Özgan for the (as Gurban would say) *semi*-professional photo setup on my examination day and thanks to Dr. Adrian Morariu, who encouraged me to try out a research path at the Intelligent Systems Group. I would also like to thank Marion Handke, who always kept an eye on any organizational aspect for me. I am also thankful for all the chats in the hallway with Leonid Lorenz, Bernd Holzke, and Dr. Jörg Petersen. Furthermore, I had the pleasure of teaching and working with countless talented students, whom I would like to thank for their commitment and enthusiasm, most notably Robin Peretzke, Robin Müller, Samuel Matthew Koesnadi, Jonas Müller, Neelu Madan, and Falko Heitzer. It was a pleasure working with all these people for the last couple of years.

A further very warm thank you goes to my whole family and friends who supported me all the way. Special thanks to my brothers Van Thien Pham and Tien Duc Pham for taking care of me and keeping me grounded when growing up together and even nowadays.

I am deeply grateful to my parents, Thi Kim Ngoc Tran and Tien Dung Pham. They risked their lives and gave up everything they knew and had in Vietnam, only for their children to have a safe childhood in a country unknown to them. I can only imagine the sorrow and frustration of having no other option than leaving behind their homes, families, and everything dear to them.
Without their incredible sacrifice, I would never have had the chance to pursue the opportunities I was lucky enough to encounter, including this journey towards a doctoral degree. *Con cảm ơn ba má!*

And of course all of this would by far not have been possible without the continuous encouragement, patience, understanding, and so very much more from my soulmate and loving wife.
Thank you so much, Kim! *For everything.*

Duc Duy Pham

Abstract

In this dissertation, novel ideas regarding the usage of prior anatomical information in fully automated image segmentation pipelines are presented and investigated. In the context of traditional segmentation methods, primitive shape priors are used to construct contour initialization methods, which complement traditional contour based segmentation approaches towards full automation. In the scope of this thesis, these initialization methods, namely Polar Appearance Models (PAMs) and Gradient based Expanding Spherical Appearance Models (GESAMs), are specifically designed for the extraction of the femoral bone in MR volumes.

Regarding deep learning, full automation is already implied by the architectural end-to-end design of fully convolutional segmentation networks. Their performance can, however, be increased by sufficient incorporation of prior anatomical knowledge. In regards to shape priors, a cascaded convolutional distance transform is proposed, which directly integrates the distance transform, as a conventional representation for shape, into arbitrary segmentation networks. Moreover, two imitating encoder based architectures are introduced, in which the compressing property of convolutional autoencoders is leveraged to infuse shape information during training. Furthermore, their applicability in cross-modality and one-shot settings is demonstrated. In case of zero-shot domain adaptation, three strategies, i.e. shape priors by Oktay et al.'s ACNN [OF+18], contour infusion by edge enhancement, and feature abstraction by color augmentation, are introduced in this specific setting, all enforcing shape aware feature learning to gap the domain shift to unseen target domains.

Additionally, a novel deep learning segmentation approach is presented for small structures with strong shape variations, which considers topographical priors by means of multitask learning. A similar topography aware approach is shown in an excursion to weakly supervised caries detection in smartphone images.

The insights from both shape and topology based deep learning architectures are combined in an application pipeline for the projection of necrotic tissue from MR volumes onto fluoroscopic x-ray images. In this scope, a procedure is presented to extract landmarks of the femoral bone, which are used in an evolution strategy to find a suitable projection.

The dissertation is concluded with a discussion about prospects and limitations of the proposed approaches for future research.

Kurzfassung

In der vorliegenden Dissertation werden neue Ideen bezüglich der Nutzung von anatomischen Priors, d.h. anatomischem Vorwissen, vorgestellt, die in vollautomatisierten Bildsegmentierungspipelines Anwendung finden sollen. Im Zusammenhang von traditionellen Segmentierungsverfahren werden primitive Shape Priors verwendet, um Methoden zur Konturinitialisierung zu entwickeln, die klassische konturbasierte Segmentierungsverfahren in Richtung Vollautomatisierung ergänzen. Im Rahmen dieser Arbeit werden die Initialisierungsverfahren Polar Appearance Models (PAMs) und Gradient based Expanding Spherical Appearance Models (GESAMs) speziell für den Anwendungsfall der Femurextraktion aus MRT Volumen entwickelt.

Im Deep Learning Kontext sind Fully Convolutional Segmentierungsnetzwerke in der Regel aufgrund ihrer End-to-End Architektur bereits vollautomatisiert. Eine Segmentierungsvorhersage kann aus einem Eingabebild erzeugt werden, ohne dass weitere Schritte zwingend erforderlich sind. Anatomische Priors sollen in diesem Zuge zur Verbesserung der Segmentierungsqualität beitragen, indem sie sinnvoll in die Netzwerkarchitektur oder den Trainingsprozess eingebettet werden. Bezüglich Shape Priors wird eine kaskadierte, faltungsbasierte Distanztransformation zur Formrepräsentation vorgestellt, die direkt in beliebige Segmentierungsnetze integriert werden kann. Des Weiteren werden zwei Architekturen präsentiert, die auf imitierenden Encodern basieren. Diese nutzen die komprimierende Eigenschaft von Convolutional Autoencodern aus, um Forminformationen während des Trainings in das Netzwerk zu injizieren. Ihre Eignung für die Anwendung in Cross-Modality und One-Shot Szenarien wird zudem demonstriert.

Weiterhin werden im Rahmen der Zero-Shot Domain Adaptation drei Strategien eingeführt, die formbasiertes Feature Learning forcieren, um den Domain Shift zu noch unbekannten Zieldomänen zu überwinden. Speziell werden Netzwerke durch Shape Priors nach Oktay et al.'s ACNN [OF+18] ergänzt, Konturinformationen durch Kantenhervorhebung stärker berücksichtigt und eine Featureabstraktion durch die Augmentierung von Farben erzwungen.

Zusätzlich wird eine neue Deep Learning Architektur vorgestellt, die kleine Strukturen mit starker Formvariabilität extrahiert, indem sie topographische Priors mithilfe von Multitask-Learning berücksichtigt. Ein ähnliches Verfahren, das ebenfalls topographisches Vorwissen nutzt, wird im Rahmen einer Exkursion zu schwach überwachten Detektionsverfahren zur Kariesidentifikation in Smartphonebildern vorgestellt.

Die Erkenntnisse bezüglich form- und topographiebasierten Deep Learning Architekturen werden in einer Applikationspipeline zusammengeführt, die sich mit der

Projektion von nekrotischem Gewebe aus 3D MRT Volumen auf 2D Röntgenbilder befasst. In diesem Rahmen wird ein Verfahren zur Extraktion von Orientierungspunkten des Femurs vorgestellt, die in einer Evolution Strategy genutzt werden, um eine geeignete Projektion zu finden.

Die Dissertation wird mit einer kurzen Diskussion über Perspektiven und Einschränkungen der vorgestellten Verfahren für zukünftige Forschungsarbeiten abgeschlossen.

Contents

Chapter 1

Introduction

Medical imaging is an important technique in clinical diagnostics, which allows medical professionals visual insight to structures within the human body without any need for invasive intervention. With the rapid technological development and the present computational capabilities, medical imaging has experienced an immense evolution from analog two-dimensional x-ray projections towards digital multi-dimensional and high resolution images of various modalities, such as Computer Tomography (CT) and Magnetic Resonance (MR) Imaging. These domain specific medical images are used for further interpretation and analysis by domain experts.

One crucial task of medical image analysis is the segmentation of the image into semantically coherent areas. The task often consists of segmenting the image into a *foreground* and *background* area of a specific structure of interest in order to extract it. This allows the generation of patient-specific models for simulations and further diagnostics. Since manual segmentation is expensive and time consuming, research on automated approaches has sustained in the last decades. Before the emergence of popular deep learning based segmentation methods, *traditional* segmentation methods have been predominantly used. These are arguably still of practical relevance, particularly when not enough suitable annotated training data is available. A major limitation in these methods is often the missing initialization stage, as for many traditional segmentation methods, a rough initial contour needs to be manually positioned by the clinician. In some cases, this is difficult to accomplish or time-consuming, e.g. in 3D segmentation methods. Automated initialization approaches would complement these methods towards full automation.

Deep learning based segmentation methods, on the other hand, already provide a complete segmentation pipeline, as most of them are designed in an *end-to-end* manner. This means, that the clinician uses a medical image as input and the deep learning based approach returns a segmentation proposal without the need for any further intervention. Of course, the results could be post-processed to correct the segmentation proposal, but this is not an indispensable step, which is required for full automation. These kind of segmentation approaches are additionally very fast during *inference*, i.e. after training them, and demonstrate state of the art performance in many application domains. Although Isensee et al. [Ise+21] show that a correctly configured commonly used neural network, i.e. Ronneberger et al.'s **U-Net** [RFB15], is able to achieve impressive scores in many segmentation challenges, research towards better segmentation architectures remains a relevant topic. It needs

to be noted, that when using a comparably trained baseline under similar circumstances, ablation studies often give more insights to the performance improvements, than comparisons to well performing models, for which the training environments are unclear.
Furthermore, the applicability of so-called *one-shot* techniques, in which only one sample is used for training, is still an important research question, since annotated data, especially in the medical context, is scarce and difficult to acquire.
Moreover, strategies to adapt a trained deep learning method towards a different domain, i.e. domain adaptation approaches, are also still active research areas in the context of medical image segmentation, as these methods help in circumventing the repeated acquisition of training data for different application domains.

1.1 Motivation

The main goal of this dissertation is to investigate, how the concept of prior anatomical knowledge can aid in making advancements in the aforementioned research areas. This thesis leverages the observation, that in many medical images various anatomical properties stay consistent both across patients and imaging modalities.

The first considered anatomical property is the prior knowledge about the shape of bones and organs. The possibility to reduce certain bones to primitive shapes, e.g. circles or spheres, is beneficial for the construction of contour initialization methods, that can be used to complement traditional fully automated segmentation pipelines. Regarding deep learning methods, incorporating shape information into the training procedure, may improve the general segmentation performance. To accomplish this, suitable shape infusion methods need to be developed, first. Shape incorporation may also be advantageous in the context of one-shot learning, as additional constraints regarding shape may restrict outlier predictions. A further possible application area for shape priors is the field of domain adaptation. Here, intensity values can differ drastically between training and application domain, therefore an abstraction from intensity or texture-based feature learning towards shape aware features is desirable. Particularly the field of zero-shot domain adaptation or domain generalization, in which no information about the target application domain is available, currently lacks research contributions.

The second considered anatomical property is the topographical location of specific structures. Some structures may show a very high variation in their shape, e.g. necrotic areas. Therefore, shape aware methods may not be applicable for the extraction of these structures. However, they are often restricted within a specific topographical region. Incorporating this prior knowledge into the training process of deep learning models, could also aid in avoiding outlier predictions, which are not even close the desired structure of interest.

1.2 Contributions

The contributions of this dissertation are the following:

- Two novel contour initialization methods for the segmentation of the femoral bone in MR images are presented [Pha+18; Pha+19c]. Both contribute to the completion of traditional fully automated segmentation pipelines.

- Regarding shape incorporation into deep learning networks, a novel deep learning component is introduced, using a cascaded convolutional distance transform to incorporate shape information [PDP21a].
- Furthermore, a novel imitating encoder - enhanced decoder design is presented, which compresses shape information from ground truth segmentation maps and enforces a similar compression for medical input images [Pha+19a].
- This architecture is extended with an existing shape infusing method and applied in a cross-modality learning setting, in which multiple imaging modalities are used for training at the same time [Kav+21].
- Moreover, the performance of the proposed extensions is investigated in the context of one-shot segmentation [PDP21b].
- In the context of domain generalization, strategies to enforce shape aware features are presented [PDP20], which significantly improve the segmentation performance on unseen application domains.
- A further contribution is made regarding the incorporation of topographical priors by proposing a multi-task learning architecture on the example of avascular necrosis extraction within the femoral head [Pha+20].
- Finally, a necrosis projection pipeline is suggested, which extracts necrotic tissue from MR volumes and projects it to an x-ray image for surgical navigation. The methods in this pipeline are partly based on the aforementioned contributions.

1.3 Outline

The remainder of this thesis is structured as follows:
In chapter 2, the fundamentals regarding the considered imaging modalities are shortly presented. Furthermore, relevant traditional segmentation methods are briefly described, and a short introduction to the development and functionality of neural networks is given.
In chapter 3, two initialization methods for traditional segmentation approaches are presented for the extraction of the femoral bone from MR volumes. Their performance is compared to each other and to a suitable registration baseline approach. Since both methods have multiple hyper parameters, these are analyzed in a retrospective manner. By the end of this chapter, all considered initialization methods are incorporated into a Level Set based segmentation pipeline.
Chapter 4 deals with the proposals of multiple end-to-end deep learning segmentation strategies, that incorporate shape priors into the learning process. The applicability of selected strategies additionally investigated in the context of cross-modality learning and one-shot segmentation. One strategy is, however, specifically designed for the application in domain generalization settings.
In chapter 5 an architecture design for topographical priors is presented. Furthermore, an excursion towards weakly supervised caries detection using topographical priors is made.
Chapter 6 combines the concepts of chapters 4 and 5 to address the practical task of necrosis projection from MR volumes to x-ray images using evolution strategies.
The dissertation finishes with chapter 7, in which a conclusion is drawn and possible future research is discussed.

Chapter 2

Fundamentals

Medical image analysis is a broad research field that generally deals with the extraction of knowledge from analyzing biomedical images of different modalities. It consists both of manual analysis, in which domain experts use images for staging or for the extraction of clinically relevant measurements, and of automated processes, in which image processing and machine learning methods are employed for tasks such as classification, anomaly detection, object localization, and image segmentation.
This chapter provides fundamental information about the kind of medical image modalities, which are considered in the scope of this thesis. Moreover, it introduces image processing and deep learning approaches, which are used in subsequent chapters for the accomplishment of fully automated image segmentation pipelines. In the scope of this thesis, image segmentation approaches are differentiated between *traditional* and *deep learning based* methods. While deep learning based techniques are relatively new and make use of deep artificial neural networks, strategies that have been established outside the domain of deep artificial neural networks are considered traditional.

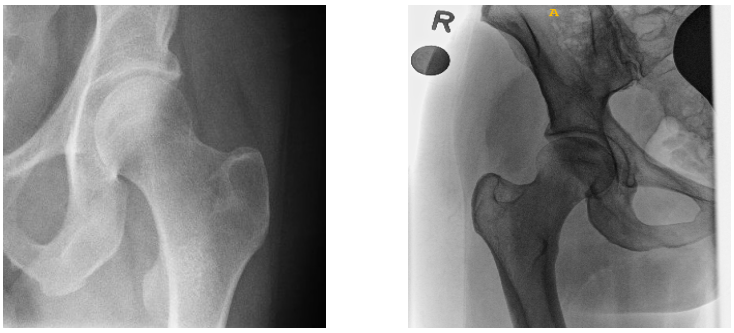
2.1 Medical Images

In clinical examinations, various imaging techniques are used to facilitate the diagnosis of specific diseases. In this section, the considered image modalities are shortly described and the general notation regarding image representation throughout the remainder of this thesis is presented. The content regarding image modalities is based on Aumüller et al.'s teaching book [Aum+14].

2.1.1 X-ray Images

One of the earliest medical imaging modalities are x-ray images. X-rays were discovered at the end of the 19th century, making it possible for the first time to visualize internal body structures without surgical intervention. In x-ray images, high-energy electromagnetic waves, i.e. x-ray radiation, are directed towards the patient, such that dense three-dimensional structures are projected onto a two-dimensional image. The principle of this procedure is based on the absorption and scattering behavior of the atoms in human tissue, induced by x-ray radiation. Tissue composed of elements of higher atomic order absorbs radiation better than tissue composed mostly of elements of lower atomic order. For example, bone tissue, which is very

rich in calcium (atomic order of 20), absorbs radiation better than lung tissue, which is largely composed of hydrogen (atomic order of 1) and carbon (atomic order of 14). As a result, bones appear very bright in analog x-ray images, whereas soft tissue is depicted darker, since more x-ray radiation is able to pass through the tissue, blackening the hit image areas on the image carrier. The appearance is however dependent on the used image carrier. For instance, in fluoroscopic x-ray imaging the same technique is used, but a digital radiation detection plate is used instead of an analog image carrier. In these kind of x-ray images, bone projections appear darker, as in these areas fewer radiation is detected on the detection plate. This technique can particularly be used for orientation and navigation during surgery, in which multiple x-ray images are taken subsequently to track the position and orientation of invasive surgery instruments. An comparison of the different intensity appearance is shown in Fig. 2.1 on two exemplary x-ray images of the proximal femur. In the remainder of this thesis, x-ray images will be simply referred to as *X-rays*, independent of the used image carrier.



(a) (b)

FIGURE 2.1: Exemplary x-ray images of the proximal femur with different intensity appearance. (a) X-ray with bright bones [Ul]. (b) X-ray with dark bones from inhouse data set.

2.1.2 CT

With the progressive development of more powerful computer systems, x-ray technology evolved into what is known as computed tomography (CT). In CT, fixed x-ray emitters and detectors rotate around the vertical body axis of a lying patient. In the process, the transmitted x-rays are detected by the detectors on the opposite side, such that transverse sectional images can be generated with the measured signals. In contrast to conventional x-ray imaging, this method allows three-dimensional detection of the patient's internal structures. In CT volumes bones are depicted brighter, soft tissue darker, and water or air are shown black, as can be seen in Fig. 2.2. Even though a very high resolution can be achieved with CT, a major disadvantage lies in the radiation exposure to which the patient is subjected during the examination.

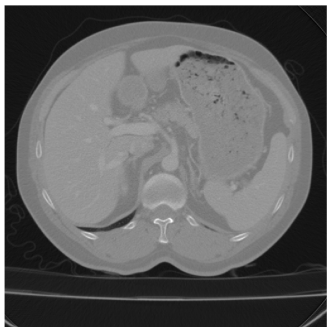
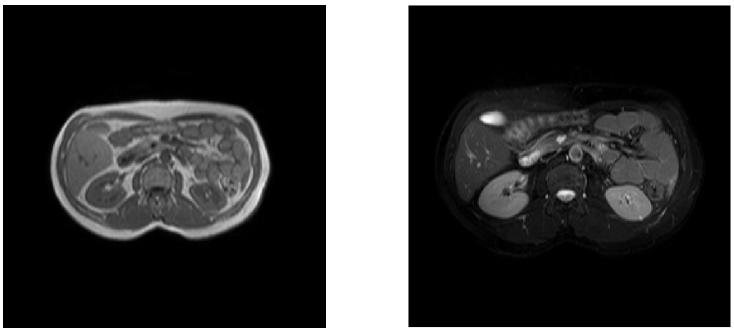


FIGURE 2.2: Exemplary CT slice of the abdomen, taken from the *CHAOS* data set [Kav+21].

2.1.3 MRI

In magnetic resonance imaging (MRI), similar to CT, the internal structures are also displayed in 3D. Compared to CT, the image acquisition takes a longer time, in which the patient needs to hold the position to avoid moving artifacts. A significant advantage of this procedure is however that the patient is not exposed to any harmful radiation, as the operating principle of MRI is fundamentally different from that of CT. In MRI, a strong magnetic field is generated that affects the so-called nuclear magnetic resonance of hydrogen atoms in the patient's tissue. The nucleus of the hydrogen atoms, each consisting of a proton, aligns itself within the magnetic field along the magnetic field lines, being in an energetically low state. If a high-frequency radiation in the radio frequency range interacts briefly with this system as a pulse, the energy level of the protons is raised for the period of the irradiation and the nuclear orientation is changed. When the pulse is terminated, the protons return to the more favorable energy state and realign with the field lines. The return to the lower energy level from the excited state is called *relaxation* and induces a voltage that can be measured. The relaxation behavior of hydrogen nuclei depends on the molecule in which they are embedded. Thus, each tissue has a specific voltage signal that can be measured upon relaxation. Relaxation can be roughly divided into T1-weighted and T2-weighted relaxation, resulting in different appearances of the generated volume. Different configurations, which affect the final appearance are called *sequences*. In T1-weighted MRI volumes, water is shown dark and fat is shown light, whereas in T2-weighted images, water is shown light and fat is shown gray. Fig. 2.3 illustrates how weighting affects appearance using the example of an MRI slice of the abdomen. In Fig. 2.3 (a) a T1-weighted image is shown, where fat tissue, e.g. under the skin, is depicted bright and water, e.g. the cerebrospinal fluid within the vertrebal canal is visualized in black. In the T2-weighted image in Fig. 2.3 (b), the cerebrospinal fluid is shown light and the fat tissue is almost black.

In general, CT and MR volumes can be considered as a stack of two-dimensional images, which will be denoted as *slices*. An important characteristic value of the generated volume set is the *Field of View (FOV)*. This describes the height and width of the field that can be captured in a slice. Thus, with a large FOV, larger body regions are covered, whereas with a small FOV, smaller regions with finer structures can be assessed. Another quantity is the slice thickness, which indicates how much distance there is from one slice to the next.



(a) (b)

FIGURE 2.3: Exemplary T1 (a) and T2 (b) weighted slice of the abdomen, taken from the *CHAOS* data set [Kav+21].

2.1.4 General Image Representation

In the scope of this thesis, the terms *image* and *volume* will be used synonymously, since a volume can be considered a 3-dimensional image. The same goes for the terms *pixel* and *voxel*. An n-dimensional image is formalized as a mapping

$$I : \Omega \to \mathbb{R}$$

from the index space $\Omega \subset \mathbb{R}$ to the space of real valued intensities. Thus, $I(p)$ returns an intensity value at the pixel position $p \in \Omega$. In practice, n-dimensional images are realized as tensors, in which an intensity value is stored for each index position $p \in \Omega$.

2.2 Image Segmentation

In medical image analysis, *image segmentation* is a crucial task for the extraction of patient-specific structures for simulations and further diagnostics. Image segmentation describes the partitioning of an image into non-overlapping (often semantically coherent) areas, each belonging to a specific class. Regarding of the extraction of a certain structure, the image is said to be segmented into the foreground and background areas of the structure of interest. Here, the foreground area represents the particular structure of interest, whereas the remainder of the image belongs to the background. In the scope of this thesis, the terms *extraction* and *segmentation* of a specific structure are therefore used synonymous.

In segmentation tasks, a desired *ground truth* is usually needed at least for evaluation. In the context of this thesis, the ground truth is represented as a binary tensor. Its size is extended by an additional *channel* dimension to account for the class assignment of each pixel. Let $\mathscr{C} := 0, 1, \ldots, C-1$ denote the set of C classes, a pixel can be assigned to. With the ground truth tensor, it is then possible to check, whether a pixel p belongs to class $c \in \mathscr{C}$ by means of the binary value at image position p and channel position c, i.e. $GT(p, c)$.
For a foreground/background segmentation task, the ground truth would thus be realized as a tensor with 2 channels.
Formally, the ground truth is therefore represented as a mapping

$$GT : \Omega \times \mathscr{C} \to \{0, 1\}.$$

For the remainder of this thesis, the notation $GT^{(c)}(p)$ is used equivalently to $GT(p, c)$. In the following subsections, two traditional segmentation approaches are presented, that are utilized within this thesis, either as baseline or as a component of a further strategy. Afterwards, the evolution of artificial neural networks is briefly presented, leading to the description of deep learning based segmentation methods.

2.2.1 Atlas Registration

The first family of traditional segmentation methods are *atlas registration* methods. Here, the general idea is to find a transformation from a *moving* image to a *fixed* image, such that the transformed image is aligned to the fixed image. Let I_m denote a moving image with corresponding ground truth GT_m and let I_f be the fixed image to be segmented. Then, the tuple (I_m, GT_m) is called an *atlas*. The aim of atlas registration methods is to find a suitable transformation between images

$$T : \mathscr{I} \to \mathscr{I},$$

where $\mathscr{I}$ denotes the space of all images. This transformation can then be used on the ground truth of the moving image for an estimate of the segmentation of I_f. The transformation T is acquired by minimizing a cost term $\mathscr{L}(I_f, T(I_m|\Theta))$, which considers the differences of the transformed moving image $T(I_m|\Theta)$ to the fixed image I_m, given the transformation parameters Θ, i.e.

$$\Theta^* := \underset{\Theta}{\operatorname{argmin}}\{\mathscr{L}(I_f, T(I_m|\Theta))\}.$$

A simple cost term is e.g. the L2-norm, i.e.

$$\mathscr{L}(I_f, T(I_m|\Theta)) := ||I_f - T(I_m|\Theta)||_2.$$

The final segmentation map, acquired by a single atlas registration, would then be calculated as $T(GT_m^{(c)}|\Theta^*)$ for all $c \in \mathscr{C}$. Atlas registration methods can differ in the type of applied transformations (rigid or non-rigid) and in the definition of the cost term, which rates the similarity of transformed moving image and fixed image.
For instance, the transformation can be either applied on the the whole moving image or only on fewer landmark points. Therefore, the cost term can utilize intensity based similarity measures or landmark based distance measures. If multiple atlases are available, a *multi*-atlas registration approach, which combines the results of multiple atlas registrations, usually achieves more robust predictions. *MATLAB* provides several different registration approaches and *Elastix* is a commonly used toolbox, which offers a wide range of registration methods.

2.2.2 Level Set

The family of *Level Set* methods is based on Kass et al.'s snakes active contour models [KWT88] and has been first described by Osher and Sethian [OS88]. The general idea is to approximate the contour of the structure of interest by iteratively deforming an initial contour. Kass et al. describe the contour by means of *snakes*, which are represented by parameterized closed curves. The deformation is accomplished by minimizing an energy functional, which is dependent on the current state of the curve and comprises internal, external, and image dependent energy. Let

$$\Gamma : [0, 1] \to \mathbb{R}^n$$

be a parameterized curve in $\mathbb{R}^n$ with parameter $s \in [0,1]$. Then the total energy functional is

$$E_{total} := \int_0^1 E_{int}(\Gamma(s)) + E_{ext}(\Gamma(s)) + E_{img}(\Gamma(s))\,\mathbf{d}s,$$

where E_{int} denotes the internal energy, E_{ext} the external energy, and E_{img} the image dependent energy. The internal energy describes properties of the curve itself, such as elasticity or curvature, whereas the external energy considers the influence of a user. In case that user influence is not desired, except for the contour initialization, the term external energy is often used synonymous to image dependent energy, which encourages the contour propagation towards specific image features, such as edges.
In contrast to snakes, Level Set methods are parameter free, which comes with major benefits. Level Sets allow a simple representation and implementation of higher dimensional contours. Additionally, topological changes like the separation of one contour to multiple contours, are canonically supported. Let

$$I : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$$

be an n-dimensional image, where Ω denotes the index space. In the following $\hat{\Gamma} \subset \Omega$ denotes the set of any point in Ω belonging to the curve Γ and will be referred to as *contour* for simplicity. In case of Level Set methods, a n-dimensional embedding function

$$\Phi : \Omega \rightarrow \mathbb{R}$$

is defined, such that the intersection of its graph and the zero-level hyperplane results in the contour, i.e.

$$\{p \in \Omega | \Phi(p) = 0\} = \hat{\Gamma},$$

where $p \in \Omega$ is an n-dimensional index point. Therefore, the following equality holds for all points on the curve Γ:

$$\Phi(\Gamma(s)) = 0 \quad , \forall s \in [0,1].$$

By means of the embedding function, the curve Γ can be implicitly defined without the necessity for any parameters $s \in [0,1]$. The embedding function is often defined as the signed distance function from the current contour, i.e.:

$$\Phi(p) := \begin{cases} -\min\limits_{q \in \hat{\Gamma}} ||p-q|| & \text{, if } p \text{ is enclosed by } \hat{\Gamma} \\ \min\limits_{q \in \hat{\Gamma}} ||p-q|| & \text{, if } p \text{ is excluded by } \hat{\Gamma} \\ 0 & \text{, else.} \end{cases}$$

The initial contour must be however known, e.g. by a tensor representation of $\hat{\Gamma}$ to initialize Φ. Figure 2.4 illustrates that the intersection of the embedding function Φ and the zero level results in the contour of interest. Any points outside the contour have a value $\Phi(p) > 0$, whereas any points within the contour yield $\Phi(p) < 0$. For points, which are exactly on the contour, the embedding function returns $\Phi(p) = 0$. Any contour deformation can be therefore accomplished by changing the appear-

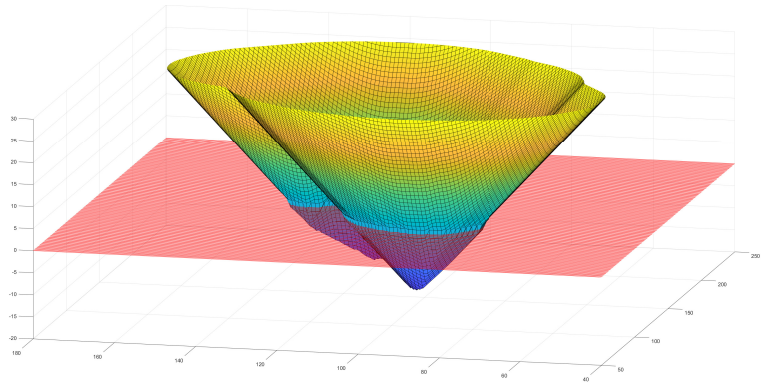


FIGURE 2.4: Illustration of intersection of zero level with embedding function.

ance of the embedding function. Instead of defining the energy functional based on the curve Γ, it can be defined based on the embedding function Φ instead. To represent the state of Φ in time, it is denoted as Φ^t to indicate the additional time component. Given an energy functional E, the embedding function is updated using gradient descent (see section 2.3.1 for more details)

$$\Phi^{t+1} := \Phi^t - \nabla_{\Phi^t} E,$$

where $\nabla_{\Phi^t} E$ denotes the gradient of E with respect to Φ^t. Depending on the problem domain, various energy functionals can be defined, which are in general edge or region dependent. In the scope of this dissertation, an energy functional based on gradient vector flows (GVF), introduced by Xu et al. for snakes [XP97] and adapted by Paragios et al. [PMGR01] to the Level Set framework, is used as traditional segmentation method in chapter 3. Moreover, a simplification of Chan and Vese's region based energy functional [CV01] is utilized in the proposed method of chapter 3.3.

2.3 Artificial Neural Networks

Image segmentation is the process of assigning each image pixel a semantic class. Therefore, machine learning approaches are suitable data driven methods to address this pixel-wise classification task. In recent years, especially *deep learning* methods have experienced a renaissance from the 80's. With Krizhevsky et al.'s [KSH12] introduction of *AlexNet*, yielding superior performance on the ImageNet [Den+09b] large scale image classification challenge, the application domain of artificial neural networks, particularly of convolutional neural networks, has expanded drastically beyond mere image classification. In this section, the fundamental structure and functionality of such artificial neural networks are explored, allowing a better understanding of more advanced artificial neural network architectures.

2.3.1 Evolution of Artificial Neural Networks

Initially, artificial neurons have been designed to simulate biological neurons in an attempt to better understand the human brain. The information processing mechanism within a biological neuron is illustrated in Fig. 2.5 and can be simplified as follows:

- Multiple incoming signals from other neurons are received at the neuron's dendrites. Here, the incoming signals can be either of excitatory or inhibiting nature.
- The received signals are then accumulated at the neuron's *axon hill*.
- From here, a signal is forwarded along its *axon* to its *synaptic endpoints* if and only if a certain threshold is reached by the incoming signals.
- In case of transmission, the fired signal does not contain information about the intensity of incoming signals, which is referred to as the *all or nothing principle*.
- At the synaptic endpoints the signal is finally transferred to subsequent connected neurons.

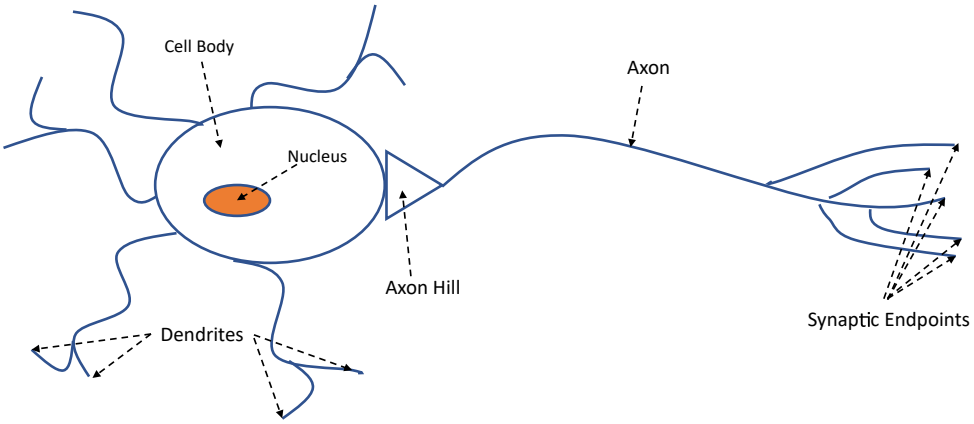


FIGURE 2.5: Simplified sketch of a neuron.

McCulloch-Pitts Cell

One of the first artificial neurons was introduced by McCulloch and Pitts [MP43]. They present a technical neuron model which artificially simulates a simplified biological behavior. Let $n \in \mathbb{N}\backslash\{0\}$ denote the number of incoming signals, then a *McCulloch-Pitts cell* (MP-cell) receives binary incoming signals $x_1, x_2, \ldots, x_n \in \{0, 1\}$, which can be either excitatory or inhibiting, like their biological correspondents. Without loss of generality, let the first $1 \leq l \leq n$ incoming signals be excitatory and the remaining $n - l$ signals be inhibiting. A MP-cell is defined to return 1 if and only if the sum of activating incoming signals surpasses a predefined threshold $\zeta \in \mathbb{R}$ and if all inhibiting signals are zero. The condition represents the accumulation and processing of incoming data from different neurons. This process is formally encapsulated by a *propagating function* f_p, i.e.

$$f_p(x_1, x_2, \ldots, x_n) := \begin{pmatrix} f_{p_1}(x_1, x_2, \ldots, x_n) \\ f_{p_2}(x_1, x_2, \ldots, x_n) \end{pmatrix},$$

where each component is defined as

$$f_{p_1}(x_1, x_2, \ldots, x_n) := \sum_{i=1}^{l} x_i$$
$$f_{p_2}(x_1, x_2, \ldots, x_n) := \sum_{i=l+1}^{n} x_i.$$

Let f_a denote the *activation function* of the neuron, that returns whether a signal is fired or not. Then the MP-cell can be represented as

$$f_a(x_1, x_2, \ldots, x_n, \zeta) := \begin{cases} 1 & \text{if } f_{p_1}(x_1, x_2, \ldots, x_n) \geq \zeta \text{ and } f_{p_2}(x_1, x_2, \ldots, x_n) = 0 \\ 0 & \text{else .} \end{cases}$$

The activation function mimics the biological neuron's all or nothing principle.
A MP-cell is depicted graphically by using edges for incoming signals, marking whether they are activating or inhibiting by a small circle, and specifying the threshold ζ, as illustrated in Fig. 2.6. Multiple MP-cells can be assembled to form a network

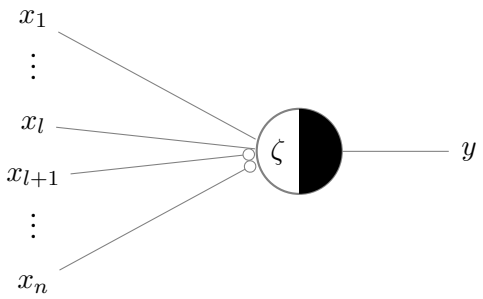


FIGURE 2.6: MP-cell with excitatory $(x_1, \ldots, x_l)$ and inhibiting $(x_{l+1}, \ldots, x_n)$ incoming signals and threshold ζ.

of MP-cells, i.e. a McCulloch Pitts Network, which can represent any binary function.

Rosenblatt Perceptron

While these initial artificial neurons do not contain any *learning* mechanism, Rosenblatt [Ros57; Ros58] proposes a similar neuron model, capable of self adaptation. The proposed *Rosenblatt Perceptron* (RP-cell) is able to process real valued inputs $x_1, x_2, \ldots, x_n \in \mathbb{R}$ and returns a scalar value, which is dependent on the activation function f_a. Each incoming signal $x_i \in \mathbb{R}$ is associated with a weight $w_i \in \mathbb{R}$ for $i = 1, 2, \ldots, n$. In Rosenblatt's formulation, the propagating function f_p is defined as the weighted sum of incoming signals, usually referred to as the *linear associator*, i.e.

$$f_p(x_1, x_2, \ldots, x_n) := \sum_{i=1}^{n} w_i x_i.$$

The activation function f_a is defined as the step function

$$f_a(x_1, x_2, \ldots, x_n, \zeta) := \begin{cases} 1 & \text{if } f_p(x_1, x_2, \ldots, x_n) \geq \zeta \\ 0 & \text{else .} \end{cases}$$

In contrast to the MP-cell the RP-cell is designed for self adaptation by adjusting the weights w_i for $i = 1, 2, \ldots, n$ and ζ. Since ζ can be considered an additional weight, that needs to be adapted, f_p and f_a can be equivalently reformulated to

$$f_p(x_0, x_1, x_2, \ldots, x_n) := \sum_{i=0}^{n} w_i x_i$$

and

$$f_a(x_0, x_1, x_2, \ldots, x_n) := \begin{cases} 1 & \text{if } f_p(x_0, x_1, x_2, \ldots, x_n) \geq 0 \\ -1 & \text{else,} \end{cases}$$

where x_0 is set to 1 and w_0 denotes the original threshold ζ, which is often referred to as *bias*. Both propagating and activation function may vary, depending on the application. For deviating definitions of f_p and f_a, the resulting artificial neuron will be simply referred to as a *Perceptron*. Fig. 2.7 shows a visualization of a generic Perceptron. Given a training set, consisting of N samples $x^{(0)}, x^{(1)}, \ldots, x^{(N-1)}$ with

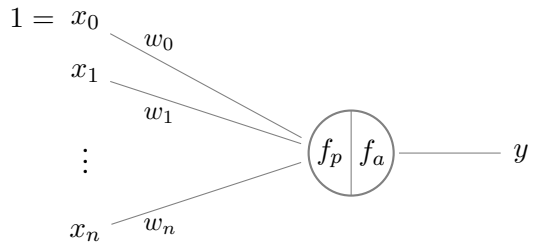


FIGURE 2.7: RP-cell with real valued incoming signals $(x_0, x_1, \ldots, x_n)$, where $x_0 = 1$ and $w_0 = \zeta$.

binary class labels $y^{(0)}, y^{(1)}, \ldots, y^{(N-1)} \in \{-1, 1\}$, a Perceptron is capable to adjust its weights, such that miss-classifications in the training are reduced if the classes are linearly separable. After weight adaptation the Perceptron can be applied to unseen data for classification. The weight adaptation process is often referred to as *learning* and can be accomplished by several learning strategies. A strategy, designed specifically for the RP-cell, is the *Perceptron Learning Algorithm* [Ros61; MP69], which requires absolute linear separability for termination. A more general method to adjust weights is the gradient descent approach, which is discussed in the following subsection.

Gradient Descent

Gradient descent is a general optimization method, which can be applied to any optimization problem, in which a differentiable cost function needs to be minimized by adapting its parameters, i.e.

$$\underset{\Theta}{\operatorname{argmin}}\, \mathscr{L}(\Theta),$$

where Θ is a parameter configuration, that effects the loss function outcome $\mathscr{L}(\Theta)$. One application example is shown in a previous section 2.2.2, where this optimization strategy is used in the context of Level Set contour propagation.
The gradient descent approach is an iterative method, in which the current weight configuration Θ^t at time t is adapted, such that the next configuration Θ^{t+1} should

be closer to the minimum of $\mathscr{L}$. It follows the idea that the gradient direction points to the next local maximum. Under the assumption that the global minimum is at the opposite direction, the next configuration is set to be

$$\Theta^{t+1} := \Theta^t - \alpha \nabla \mathscr{L}(\Theta^t),$$

where the *learning rate* $\alpha > 0$ determines the extend, in which the adaptation is applied, and $\nabla \mathscr{L}(\Theta^t)$ denotes the gradient of $\mathscr{L}$ at position Θ^t. Some drawbacks of this optimization approach are

- the possibility to get stuck in a local minimum, if the learning rate is too small
- the possibility to oversee the global minimum, if the learning rate is too large
- the limitation to differentiable loss functions

to name a few. The design of more sophisticated gradient-based and also gradient-free optimization methods is a research area for itself and is not further discussed in the scope of this dissertation.

Multi Layer Perceptron

A major drawback of using one single Perceptron is its limited capability to the classification of linearly separable tasks. Multiple Perceptrons, however, can be arranged in a layer-wise manner to a network of Perceptrons, and with the application of non-linear activation functions. These *multi layer Perceptrons* (MLPs) are also able to address non-linear classification and even regression tasks. Fig. 2.8 illustrates the

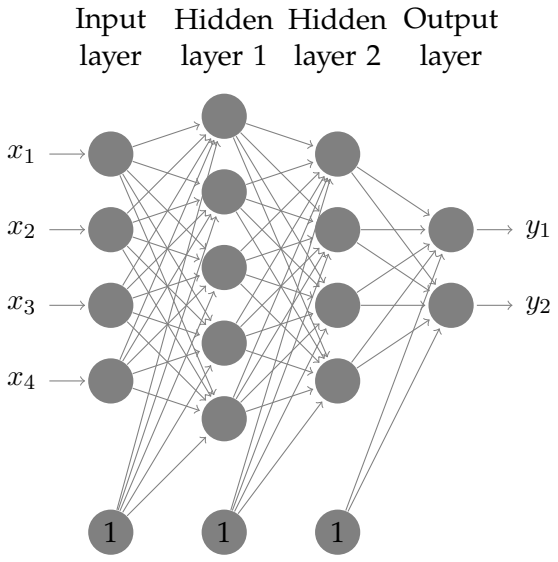


FIGURE 2.8: Exemplary scheme of a MLP.

layer-wise arrangement of multiple single Perceptrons to a MLP. The input signals are often denoted as input nodes, that do not have any associated propagation and activation function. This first layer is usually referred to as the *input layer*. The *output layer* returns the final activated output of the network, and the layers in between are denoted as *hidden layers*. While there are no connections betweeen Perceptrons within a layer, each Perceptron from a hidden or output layer receives weighted incoming signals from every Perceptron of the previous layer, including the bias

weight. Therefore, a MLP layer is often also referred to as a *fully connected (FC)* or a *dense* layer in the context of *deep* learning. A MLP is considered *deep*, if it has multiple hidden layers. The number of required hidden layers to be considered *deep* is, however, not specified in literature. As can be seen in Fig. 2.8, the number of outputs is not limited to one dimension anymore. The choice of activation function for the output layer additionally determines the task, that the designed MLP can address. Common activation functions for classification are sigmoid, tanh, and softmax, whereas the identity is often used for regression tasks. In the context of deep learning, a further commonly used activation function to enforce non-linearity is the *Rectified Linear Unit (ReLU)* function, which maps any negative value to zero and returns the identity for non-negative values. In the following, the term Perceptron and *neuron* will be used synonymously.

Backpropagation

The learning process of the MLP is accomplished by adapting the weights in between each layer by means of gradient descent. For this a differentiable loss function $\mathscr{L}$ is defined that is minimized by adapting the network weights. Let $w^t_{h,i,j}$ denote the weight connecting the i-th neuron of the h-th layer with the j-th neuron in the $(h+1)$-th layer at time t. Then, according to gradient descent, this weight needs to be adapted by

$$w^{t+1}_{h,i,j} := w^t_{h,i,j} - \alpha \frac{\partial \mathscr{L}}{\partial w^t_{h,i,j}},$$

where $\alpha > 0$ is the learning rate. The challenge in this formulation is the calculation of the differential $\frac{\partial \mathscr{L}}{\partial w^t_{h,i,j}}$. Rumelhart et al. [RHW86] describe that for the computation of the differentials of the h-th layer, it is first necessary to get the differentials for the subsequent $h+1$-th layer. Therefore, any input is first forwarded through the MLP to achieve its output. After calculating the loss $\mathscr{L}$, the differentials of the last layer are computed, which can then be used in a *backpropagation* process to get the differentials of previous layers. Finally, the calculated differentials are used to adapt the MLP weights according to gradient descent.

2.3.2 Convolutional Neural Networks

One major drawback of MLPs is the large number of needed weights to process large images. Given a 2D image of size $M \times N$, let the number of neurons in the first hidden layer be H. Then the first layer would require $(M \cdot N + 1) \cdot H$ weights. For an image with size 256×256, a MLP with only eight neurons in the first hidden layer would already need $524,296$ incoming weights. A further limitation of MLPs is the loss of positional information, as all pixels are connected to all hidden neurons. *Convolutional neural networks* (CNNs), presented by LeCun et al. [LeC+89b; LeC+89a], are a more memory efficient type of feed forward neural networks, which can also be trained by backpropagation. While MLPs consist only of Fully Connected layers, CNNs additionally comprise at least *convolutional* and *pooling* layers, which will be explained in the following paragraphs.

Convolutional Layer

The basis of a convolutional layer is the convolutional operator $*$. Given a *kernel*, denoted as κ, of size $K_1 \times K_2$, the discrete 2D convolution with a 2D image I of size

$M \times N$ is defined in a point-wise manner as

$$I * \kappa(i,j) := \sum_{k_1=0}^{K_1-1} \sum_{k_2=0}^{K_2-1} I(i-k_1, j-k_2) \cdot \kappa(k_1, k_2)$$

for all index points $(i,j) \in \Omega$ with $i = K_1-1, \ldots, M-1$ and $j = K_2-1, \ldots, N-1$. The resulting *feature map* of the convolutional operation with kernel κ shows in which image positions and to what extend the pattern, represented by the kernel κ, occurs. Thus, high deviations from zero implicate a strong response.

While convolutions have desirable mathematical properties, such as commutativity, and duality to the multiplication in the Fourier space, these traits are not relevant in the context of CNNs in practice. Instead, the discrete 2D cross-correlation

$$I \odot \kappa(i,j) := \sum_{k_1=0}^{K_1-1} \sum_{k_2=0}^{K_2-1} I(i+k_1, j+k_2) \cdot \kappa(k_1, k_2),$$

for $i = 0, \ldots, M-1-K_1$ and $j = 0, \ldots, N-1-K_2$ is used instead. Its resulting feature map $I \odot \kappa$ is practically comparable to $I * \kappa$. Following the common terminology in the deep learning community, these operations are going to be used synonymous, although it should be kept in mind that they are actually mathematically different.

In simplified terms, the kernel is *slided* over the image, beginning from the top left corner. For each position, the sum of point-wise product between kernel elements and currently overlayed image components is calculated. The result is then stored in a new array of smaller size $(M - K_1 + 1) \times (N - K_2 + 1)$ at the position, the kernel is currently overlayed at. To avoid the reduction in size, the image can be *padded* to a larger size of $(M + K_1 - 1) \times (N + K_2 - 1)$, either with zeros or with values, similar to the closest image values.

In the current formulation the point of reference, in which the result would be stored, is in the upper left corner of the kernel. Usually, the discrete 2D convolution is, however, defined, such that the point of reference is the center of the kernel. Figure 2.9 illustrates the 2D convolutional operation on a padded example. The *stride* determines, whether all pixels are considered (stride = 1) or if only every n-th pixel is considered (stride = n) while sliding the kernel over the image.

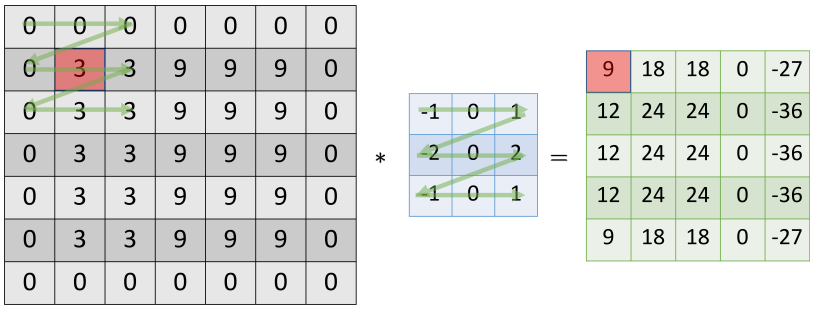


FIGURE 2.9: Discrete 2D correlation. Each component of the kernel is multiplied with the corresponding image component from the overlayed area. The sum of products is stored in a new array at the current kernel position. In this illustration the point of reference for the kernel position is in the center of the kernel.

Drawing the relationship to MLPs, each pixel position can be considered an input neuron, whereas each kernel component is considered the associated weight of the incoming signal. A major difference is, however, that by sliding the kernel over the image, the same weights are reused for multiple input nodes, whereas in MLPs each incoming signal has its own weight. This specific property of CNNs is referred to as *parameter sharing*, which drastically reduces the memory requirement for large images compared to MLPs. A further memory saving property is the *sparse connectivity*, if the kernel is chosen much smaller than the input. In this case, only small patches of the input are considered during the convolution operation. This puts every pixel into the local context of its neighborhood, instead of the whole image. The resulting feature maps indicate the location of where the feature of interest can be found in the input. This demonstrates the *translational equivariance* property of the convolution. This means, that if a particular feature is shifted in the input, the shift will become imminent in the generated feature map as well.

A *convolutional layer* consists of multiple kernels, with which its input is convolved. The idea is to have many different kernels to generate many different feature maps, as depicted in Fig. 2.10. In practice, the resulting feature maps are stored in a tensor,

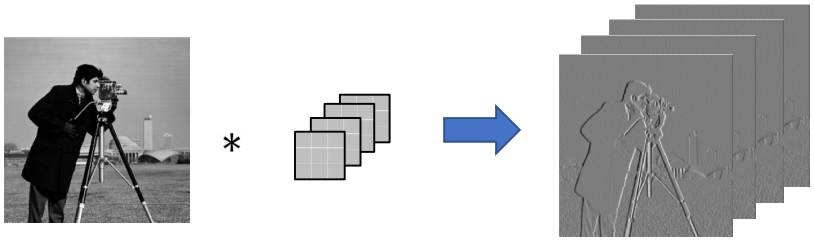


FIGURE 2.10: Convolutional layer.

where the number of feature maps determines the number of channels of the tensor. Usually the kernels are initialized randomly and adapted by means of backpropagation to converge towards meaningful filters.

Pooling Layer

A *pooling* function reduces a rectangular pixel neighborhood, which is denoted as a *pool*, to a statistical quantity. While in *max-pooling*, the maximum of the pool is returned, min-pooling and average-pooling return the pool's minimal or mean value, respectively. A *pooling layer* simply applies the pooling function on its input. Depending on the pool size, the spatial size is significantly reduced by a pooling layer. The goal of the pooling layer is to enforce translation invariance for classification tasks, which comes with a partial loss of positional information of the detected features. Often 2×2 max-pooling is performed, which is illustrated by Fig. 2.11.

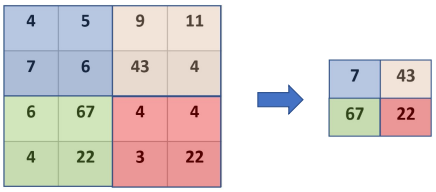


FIGURE 2.11: Max-Pooling.

For classification CNNs, multiple convolutional and pooling layers are alternately used for feature extraction purposes. While the more shallow layers learn to extract low level features, features in deeper layers show more complexity . The resulting tensor after the feature extraction process is called *latent representation*. Afterwards, FC Layers are usually used for classification, as can be seen in Fig 2.12, which shows a commonly used classification CNN, namely the **VGG-16** architecture by Simonyan and Zisserman [SZ15a]. In this case, convolutions are applied with size preserving padding, and the number of feature maps for each convolutional layer is implied by the number of channels in the resulting tensor. The pooling operation is visualized by the decreasing height and width of the resulting tensors. In this particular VGG 16-example, the numbering refers to the number of convolutional and fully connected layers, excluding the pooling layers. The output values of CNNs are usually referred to as *predictions*.

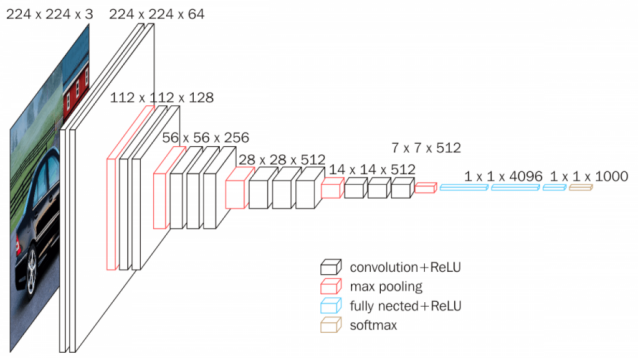


FIGURE 2.12: VGG-16 illustration, taken from [Has].

Transposed Convolutional Layer

The *transposed convolutional layer* is a layer that is able to upsample its input to a larger size by means of the *transposed convolution*. It is therefore contrary to the pooling layer. In the transposed convolution, each input component is multiplied with all kernel elements. This results in an intermediate matrix of the same size as the kernel for each input component. These intermediate matrices are inserted into a new final matrix, such that the upper left corner of the intermediate matrices are positioned at the corresponding position of the input element. Overlapping parts of the intermediate matrices are added up. The *stride* determines, whether the intermediate matrices are positioned at the exact corresponding position of the input element (stride $= 1$) in the final matrix, or whether they are positioned with an offset (stride > 1), as Fig. 2.13 illustrates.

Therefore, stride and kernel size determine the output of the transposed convolution. E.g., a transposed convolution with a kernel size of 2×2 and a stride of 2 can be used to achieve an upsampling, such that the result has the same size as the input of a pooling layer with pooling size 2×2. If the kernel additionally only consists of ones as entries, the transposed convolution would implement a nearest-neighbor upsampling approach, in which the additional intermediate rows and columns are filled with the values of the non-intermediate nearest neighbor. In contrast to common upsampling strategies like linear or cubic interpolation, the kernel values are adapted during training, resulting in a task specific upsampling method.

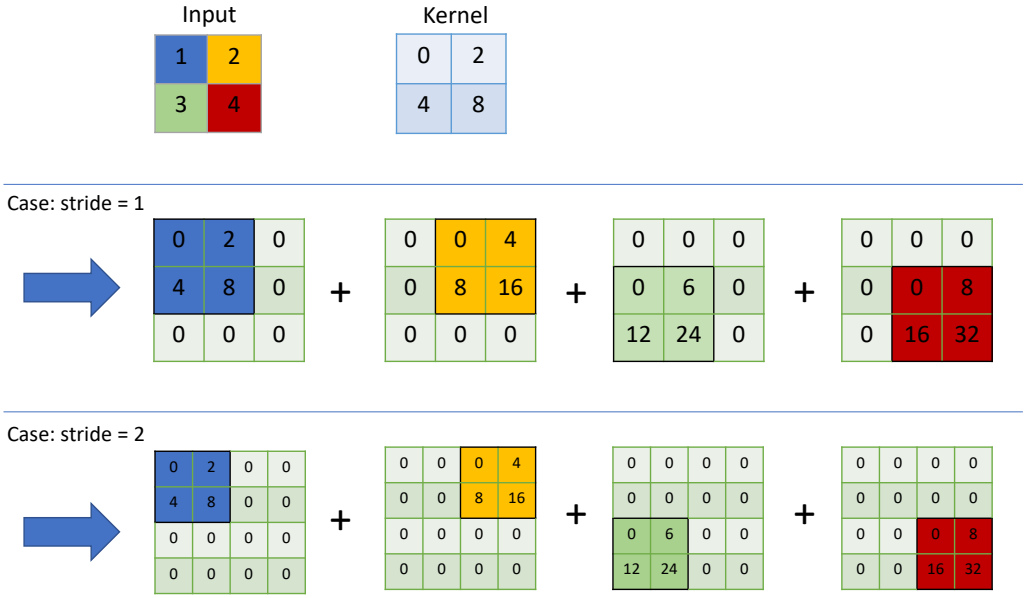


FIGURE 2.13: Illustration of the transposed convolution for two different strides. The intermediate matrices are depicted in the same color as the input components they originate from. In the first case (center row), an overlap occurs due to the small stride of 1. In case of a stride of 2 (bottom row), there is no overlap. Both scenarios showcase the influence of the stride on the output size.

2.3.3 Fully Convolutional Architectures

Originally, CNNs were designed to address classification tasks. However, the CNN structure can be modified for segmentation tasks, as shown by Long et al. with the introduction of *Fully Convolutional Networks (FCN)* [LSD15]. They use the output of the last convolutional layer and upsample it to the input size, rendering an architecture that does not make use of any FC layers, therefore being *fully convolutional*. For the upsampling process, established methods as well as transposed convolutional layers can be used. In additional variants, the feature maps from previous layers with larger scales are combined with the accordingly upsampled output of the last convolutional layer by means of *skip connections*. The fused maps are finally upsampled to the original input size, as shown in Fig. 2.14. The skip connections ensure the conservation of fine grained information of larger scale feature maps, that would be otherwise be lost in the pooling layers.

One might notice that the convolutional layers are summarized in blocks, except for the last two convolutional layers, which are depicted as *conv6* and *conv7* separately. This is due to the fact, that the last convolutional layer (i.e. *conv7*) needs as many kernels, as there are semantic classes, whereas the number of kernels for *conv6* can be arbitrary. A further peculiar naming convention may be the terms $2\times, 4\times, \ldots, 32\times$, which refer to the needed upsampling factor to match the corresponding feature map size. For instance, the resulting feature maps of *conv7* need an upsampling factor of 4 to match the size of the pooled feature maps after *pool3*, whereas the pooled feature maps after *pool4* only need a factor of 2 to match the same size. Matching sizes are required for the information fusion by means of skip connections.

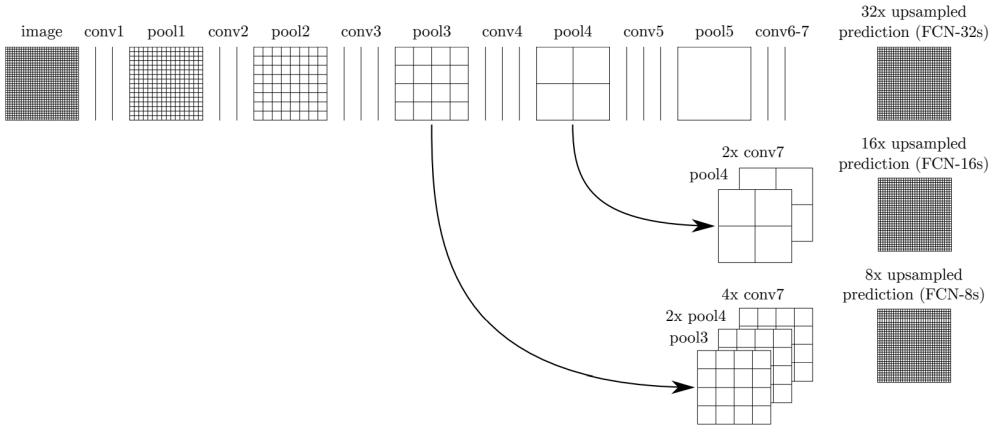


FIGURE 2.14: Illustration of a FCN and its variants with skip connections, taken from [LSD15]. Here the terms $2\times, 4\times, \ldots, 32\times$ refer to the needed upsampling factor for matching feature map sizes.

A similar CNN architecture, that has gained popularity in the medical context, is the *U-Net*, proposed by Ronneberger et al. [RFB15]. The U-Net architecture can be divided into a *contracting path* or *encoder*, which encodes the input image to a compact latent representation, and an *expanding path* or *decoder*, which decodes the compact representation to a segmentation output. In the contracting path, max pooling layers iteratively reduce the input size, rendering multiple *scale levels*. In the expanding path, transposed convolutional layers are repetitively used to regain higher scale levels. Skip connections from each scale level of the contracting path to the corresponding scale level of the expanding path ensure the consideration of fine grained information in the upsampling process. Fig. 2.15 shows the network architecture of the U-Net. In this example, 64 kernels are learned in two consecutive convolutional layers, respectively, in the first scale level. In the original publication, convolutions are not padded, thus, resulting in a slightly decreased feature map size after each convolutional layer. In the scope of this thesis, U-Nets are implemented with padded convolutions, avoiding the necessity for cropping when using skip connections.

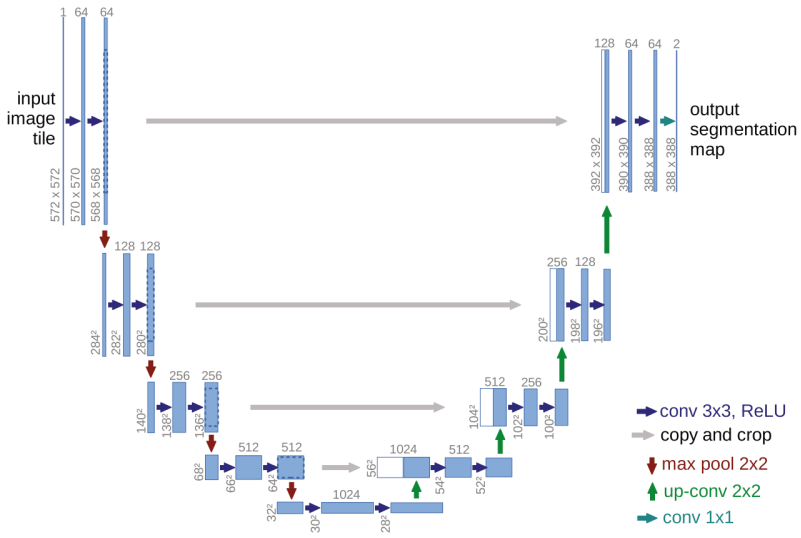


FIGURE 2.15: Illustration of the U-Net architecture, taken from [RFB15].

2.4 Evaluation Metrics

Evaluation metrics are necessary to quantitatively and objectively rate the performance of a segmentation algorithm. In accordance to the nomenclature in the deep learning context, a segmentation generated by an algorithm will be referred to as the algorithm's *segmentation prediction*, independent of whether the algorithm is traditional or deep learning based. Evaluation metrics try to quantify the quality of a segmentation prediction compared to the desired ground truth segmentation. If a domain expert visually rates a segmentation prediction as "good", this should also be reflected by the chosen evaluation metric. In the following, two evaluation metrics, that are commonly used in the context of medical image segmentation, are presented.

2.4.1 Dice Similarity Coefficient

As described in section 2.2, image segmentation is the process of assigning each image point to a class. For each image point, the presence of any class is represented by a binary value, i.e. by a *positive* or a *negative* value. For the segmentation prediction, the points classified as positive may actually belong to the object. Then they are called *true positives (TPs)*. In case of a misclassification, they are called *false positives (FPs)*. The same reasoning can be made for *true negatives (TNs)* and *false negatives (FNs)*, which is illustrated in Fig. 2.16.

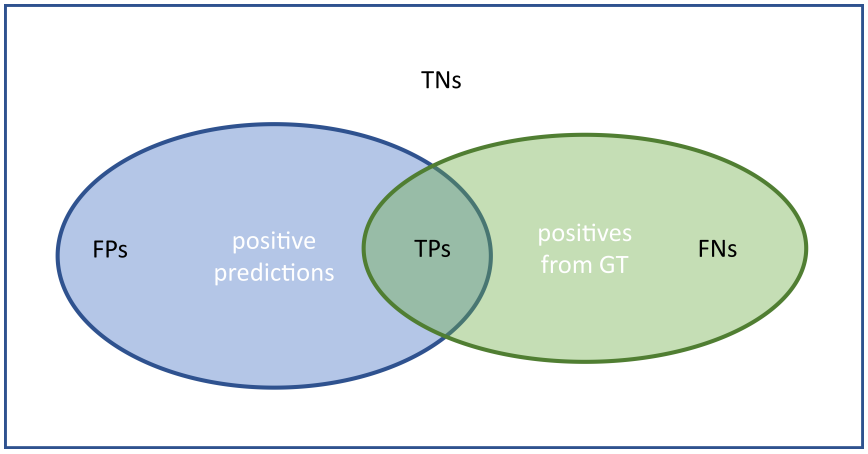


FIGURE 2.16: Illustration of TPs, FPs, TNs, and FNs.

The *precision* ($prec$) measures the ratio of the TPs to the total number of *positive* predictions, i.e.

$$prec = \frac{TPs}{TPs + FPs}.$$

It therefore is a quantification of how reliable a positive prediction of the method is. If it classifies everything as positive, the precision would expectedly be low.
However, high precision does not imply good segmentation. For example, a segmentation prediction consisting of only one pixel, classified as positive, would achieve a precision of 1, if this prediction is a true positive, since there aren't any false positives. The problem with this measure is that the FNs are not considered.

Another measure, that takes this into account, is the *sensitivity* or *recall* (rec), which is defined as

$$rec = \frac{TPs}{TPs + FNs}.$$

Recall expresses the ratio of the TPs to the total number of ground truth pixels belonging to the class of interest. Although the FNs are included in the metric, the FPs are not considered instead. Thus, a segmentation prediction, which assigns all pixels a positive value, would result in a high recall, even though the segmentation prediction is probably poor.

A widely used similarity measure that combines precision and recall is the *Dice Similarity Coefficient (DSC)*, also referred to as *dice score*. The DSC forms the harmonic mean between precision and recall. The harmonic mean of two values $x_1, x_2 \in \mathbb{R}$ is defined as

$$\overline{x}_{harm} := \frac{2}{\frac{1}{x_1} + \frac{1}{x_2}}.$$

Thus, it follows for the DSC:

$$\begin{aligned} DSC &:= \frac{2}{\frac{1}{prec} + \frac{1}{rec}} \\ &= \frac{2TPs}{(TPs + FPs) + (TPs + FNs)}. \end{aligned}$$

It should be noted, that $TPs + FPs$ yields all points predicted as positive and $TPs + FNs$ is the number of actual positive points according to the ground truth. In Fig. 2.16 this would resemble the ratio of twice the intersection area (TPs) and the summation of both the whole blue ellipse (*positive predictions*) and the whole green ellipse (*positives according to the ground truth*).

Let GT denote a ground truth tensor and $c \in \{0, 1, \ldots, C-1\}$ the class for which the DSC needs to be calculated. Furthermore, let y denote the segmentation prediction, which assigns each position and in each output channel an output score between 0 and 1. Then, the DSC for the particular class c can be calculated by

$$DSC(c) := \frac{2 \cdot \sum\limits_{p \in \Omega} GT(p, c) \cdot y(p, c) + \epsilon}{\sum\limits_{p \in \Omega} GT(p, c) + \sum\limits_{p} y(p, c) + \epsilon}, \tag{2.1}$$

where $\epsilon > 0$ is a small number to avoid dividing by zero. This results in a scalar evaluation score, that ranges between 0 and 1. A common loss function for deep segmentation networks is the mean *dice loss* over all classes, which is defined as:

$$\mathscr{L}_{dice}(GT, y) := \frac{1}{|\mathscr{C}|} \sum_{c \in \mathscr{C}} \frac{2 \cdot \sum\limits_{p \in \Omega} GT(p, c) \cdot y(p, c) + \epsilon}{\sum\limits_{p \in \Omega} GT(p, c) + \sum\limits_{p} y(p, c) + \epsilon}, \tag{2.2}$$

where $\mathscr{C}$ denotes the set of possible classes.

2.4.2 Symmetric Hausdorff Distance

While the dice score intuitively gives a good estimate of whether extracted structures overlap with the desired ground truth, the distance of misclassifications to the desired ground truth is not considered in this metric. The *symmetric Hausdorff distance (sHD)* on the other hand is a further evaluation metric for segmentation tasks, that is designed for this problem. Let A and B denote arbitrary sets, for which distances between elements from one set to the other can be calculated by a distance function

$$d_{A,B} : A \times B \to \mathbb{R}^+.$$

Then, the *Hausdorff distance* between these two sets is defined to be

$$HD(A,B) := \sup_{a \in A} \left[\inf_{b \in B} [d(a,b)] \right]. \tag{2.3}$$

The term $\inf_{b \in B} [d(a,b)]$ is however not symmetric, which is illustrated in Fig. 2.17. Given a point $a_0 \in A$, its closest point $b \in B$ may have another closest point $a_1 \in A$, which is not the same as a_0. Therefore the distance defined in Eq. (2.3) is also not

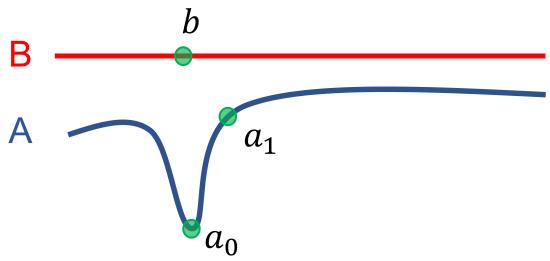


FIGURE 2.17: Illustration of non-symmetry of $\inf_{b \in B} [d(a,b)]$. b is the closest point in B to a_0. The closest point in A to b_0, however, is a_1 and not a_0.

symmetrical. To achieve this symmetry property, the symmetric Hausdorff distance is defined as

$$sHD(A,B) := max\left(HD(A,B), HD(B,A)\right). \tag{2.4}$$

Since the maximum function is symmetric, sHD is also symmetric, i.e.

$$sHD(A,B) = sHD(B,A).$$

Fig. 2.18 shows the difference of $HD(A,B)$ (left dotted line) and $HD(B,A)$ (right dotted line), which are both required to calculate the symmetric Hausdorff distance. In case that either A or B is empty (but not both), the sHD is defined to be infinity.

In the context of evaluating segmentation methods, given a class $c \in \mathscr{C}$ (e.g. the foreground class), all corresponding pixel positions assigned to c in the segmentation prediction form the first set. The second reference set is formed by all pixel positions which actually belong to c according to the ground truth. Let y denote a segmentation prediction and GT the desired ground truth, then the sHD for class $c \in \mathscr{C}$ is denoted as $sHD(y^{(c)}, GT^{(c)})$, although the sHD is actually defined for sets.

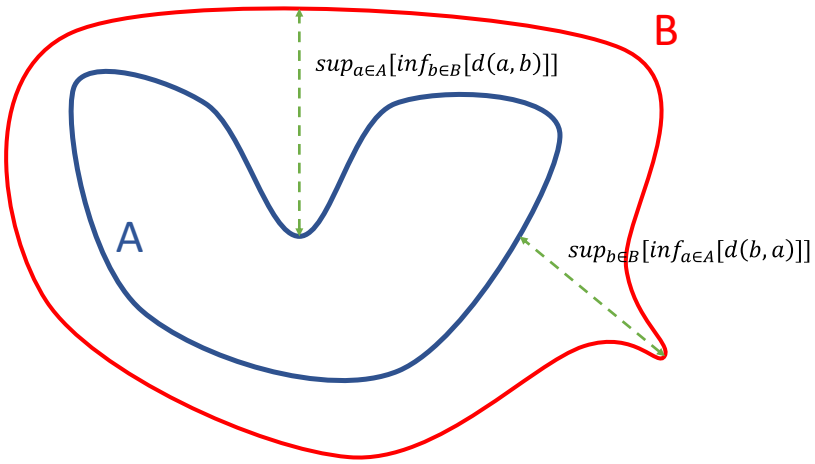


FIGURE 2.18: For the calculation of the symmetric Hausdorff distance between two sets A and B, both $HD(A,B)$ (left) and $HD(B,A)$ (right) are necessary.

Chapter 3

Primitive Shapes for Model Initialization in Femur Segmentation Pipelines

Many traditional non-deep learning segmentation strategies require a feasible initialization for a sufficient segmentation process. Particularly in contour progression based approaches, it is crucial to position the initial contour close to the structure of interest, that needs to be extracted, in order to achieve satisfactory results. These approaches comprise Level Set methods [CV01], Active Shape Models [CT92] and Active Appearance Models [CET98], in which an initial contour is iteratively adapted until it encapsulates the structure of interest. While these methods have been extended for better segmentation results in numerous ways ([Kai+09], [SMT08], [Yok+09], [Cha+14]), research regarding model initialization has been receiving reasonably less attention. In this chapter, two initialization approaches for MR images are presented to complement existing non-deep learning segmentation approaches towards full automation. The proposed initialization methods are applicable for anatomical structures that either have a mostly convex nature in the axial plane or contain a near-spherical component, respectively. In particular, the task of contour initialization for femur segmentation in three-dimensional MR images is addressed, in which the femur's shape is reduced to primitive shapes. The content of this chapter is based on the following previous publications and has been revised and adapted with permission for this chapter:

> **[Pha+18]** Duc Duy Pham et al. "Polar appearance models: a fully automatic approach for femoral model initialization in MRI". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE. 2018, pp. 1002–1005
> Copyright ©2011 IEEE

> **[Pha+19c]** Duc Duy Pham et al. "Gradient-Based Expanding Spherical Appearance Models for Femoral Model Initialization in MRI". In: *Bildverarbeitung für die Medizin 2019*. Springer, 2019, pp. 43–48

3.1 Medical Background

In modern medicine, especially in the domain of orthopedics and trauma surgery, 3D models are helpful tools to aid in preoperative planning and to design prosthetics,

tailored specifically to the patient's needs. Patient-specific models allow simulations of joint movements and the detection of possible points of friction. For these kind of simulations, first the 3D models need to be generated by segmenting the anatomical structures of interest. The human hip joint, in particular, is a structure of interest, as it carries a major portion of the body weight, therefore being naturally prone to physical deterioration.
The hip joint is a ball joint, consisting of the *femur*, that is held by a socket counterpart, the *acetabulum*. The femur can be divided into *femur head*, *femur collum*, and *femur shaft*. Further landmarks are the *minor trochanter* on the medial front side and the *major trochanter* on the lateral back side of the bone.
While CT is the predominant imaging method to visualize bones, since the acquisition process is fast and bone tissue can be clearly distinguished from the surrounding tissue, MRI has the advantage, that the patient does not need to be exposed to radiation. Therefore, research on fully automatic femur extraction strategies is an ongoing topic. This chapter especially focuses on initialization methods to complement existing traditional segmentation methods, that usually require manual model initialization.

3.2 Related Work

Regarding automated femur segmentation, there have already been several contributions. Most of these are, however, designed for X-ray images or CT volumes in which bones are reasonably more distinguishable than in MR volumes. Linder et al. [Lin+12] propose a sliding window approach utilizing Random Forest regression voting to detect the proximal femur in X-rays in order to fully automatically segment the proximal femur. Chu et al. [Chu+15] present a fully automatic hip joint segmentation approach for CT scans, that uses Random Forests for initial landmark detection and multi-atlases and Articulated Statistical Shape Models for segmentation. Kainmüller et al. [Kai+09] also present a fully automatic approach for CT scans, extending the common definition of statistical shape models to *Joint Statistical Shape Models* by additionally modeling the rotational displacement of femur to pelvis. For initialization, an extension of the generalized Hough transform is used to detect 3D objects [Kho07]. Xia et al. [Xia+13] use a multi-atlas registration method to fit the model into MRI volumes for their fully automated segmentation work flow. Tang et al. [Tan+17] make a more general contribution in presenting a deep learning approach for segmentation by integrating the Level Set model into a Fully Convolutional Network architecture, allowing the use of unlabeled training data, and leaving the need for localization of the object of interest obsolete.
Younes et al.'s approach [YNS14] first detects the structure of interest by primitive shape recognition in 3D CT data and then applies deformable shape models afterwards, which is similar to the proposed overall strategy in this section.

3.3 PAMs: Polar Appearance Models

The first initialization approach of this chapter considers a 3D MR volume as a sequence of axial 2D slices and leverages the mostly convex nature of the femur in these axial slices. It makes use of statistical appearance properties of the femur in MR images, derived from training atlases, in which both MR volumes and the corresponding desired ground truth volumes are available. The procedure can be partitioned into four stages, as shown in Fig. 3.1. The first stage comprises training an appearance model, namely the Polar Appearance Models (PAMs), using the training atlases. The PAMs are used both in the second stage to estimate the femur location in the MR volume and in the third stage in which the femur's boundary is approximated. In the last stage an Iterative Closest Point (ICP) algorithm [CM92; BM92] is applied to fit an existing generic 3D femur contour model into the approximated boundary points. Finally, the fitted model can be used as initial starting point for an subsequent segmentation approach.

3.3.1 Model Definition

The aforementioned Polar Appearance Models (PAMs) consist of two components. The first component models the intensity distribution within the femur area to roughly approximate the location of the femur within the 3D volume. The second component considers polar transformations of the axial slices, i.e. the Cartesian pixel coordinates are represented as polar coordinates, rendering a transformed image, in which the axes are defined by phase and amplitude. Here, profile lines, which are perpendicular to the actual boundary points, are used to assess intensity changes in the boundary region.

Modeling intensity distribution

Let $(I_0, GT_0), \ldots, (I_{N-1}, GT_{N-1})$ denote N training atlases, with MR volume I_i and its corresponding label volume GT_i for $i = 0, \ldots, N-1$.
As mentioned before, a 3D volume is considered a sequence of axial 2D slices. Since the femur usually does not go through all axial slices, it is necessary to determine, in which slice the femur is present. This can be easily achieved since the ground truth for any training MR volume is available.
Let M_i denote the amount of axial slices containing a fraction of the femur for the i-th atlas for $i \in \{0, \ldots, N-1\}$. For the sake of simplicity the index pair (m, i) for the m-th axial slice, $m \in \{0, \ldots, M_i - 1\}$, of the i-th atlas, $i \in \{0, \ldots, N-1\}$, will be uniquely assigned to a scalar index $k \in \{0, \ldots, K-1\}$ by

$$k := m + \sum_{l=0}^{i-1} M_l \tag{3.1}$$

for a total number of

$$K := \sum_{i=0}^{N-1} M_i$$

slices, as illustrated by Fig. 3.2. For each axial slice, the normalized intensity distribution within the femur area is computed with a fixed amount of $n_{bins} \in \mathbb{N}\backslash\{0\}$

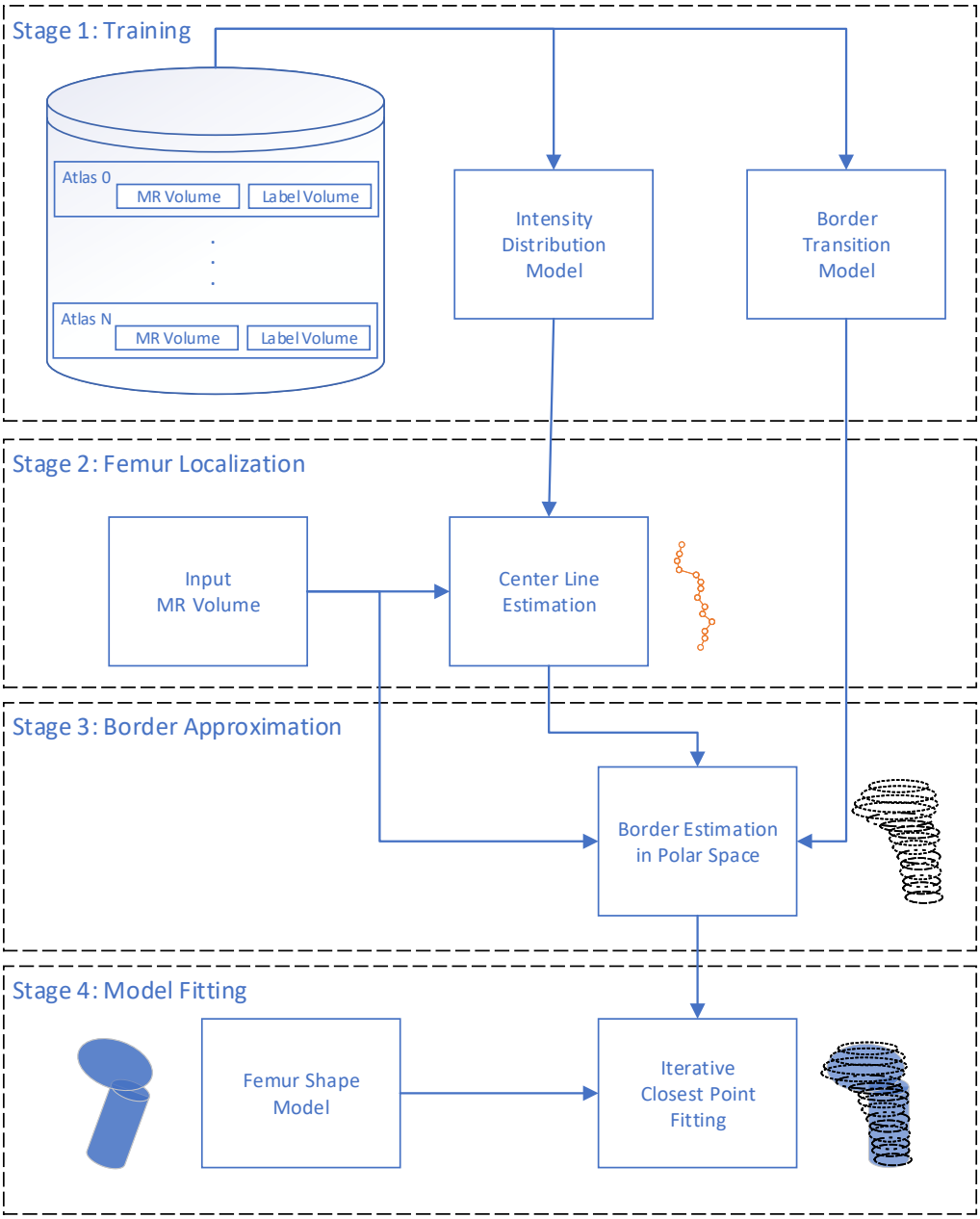


FIGURE 3.1: Overview of PAMs strategy.

intensity bins, yielding K feature vectors $v_0, \ldots, v_{K-1} \in [0,1]^{n_{bins}}$. For the calculation of the normalized intensity distribution the whole intensity range is divided into a total number of n_{bins} intensity bins $\mathscr{B}_0, \ldots, \mathscr{B}_{n_{bins}-1}$. For fixed k let ${}_kI$ denote the k-th axial slice in the whole set of axial slices, i.e. the m-th axial slice in the i-th volume I_i with $m \in \{0, \ldots, M_i - 1\}$ and $i \in \{0, \ldots, N-1\}$, such that m and i fulfill Eq. (3.1). In the same way ${}_kGT$ is defined accordingly.
Let

$$\chi_k : \Omega \times \{0, n_{bins} - 1\} \to \{0, 1\}$$

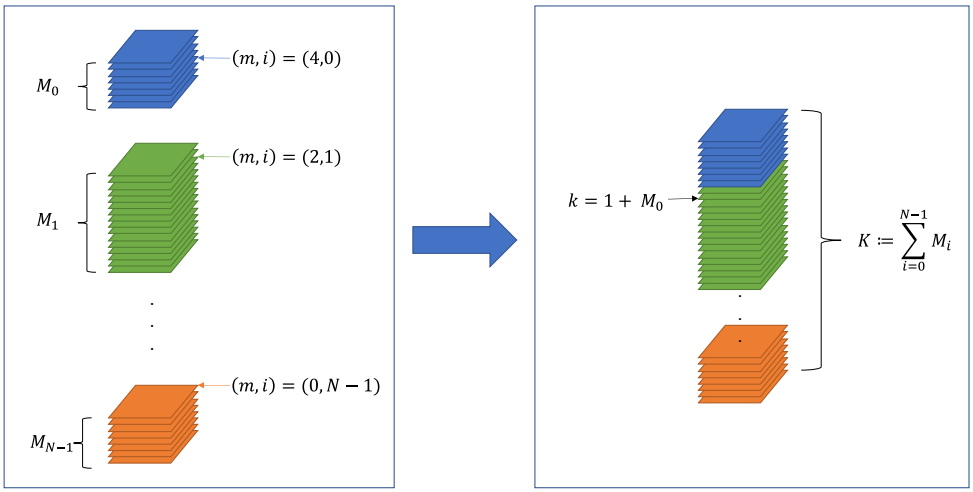


FIGURE 3.2: Simplified slice indexing. Each slice can be uniquely addressed with (m, i), where m denotes the slice index and i the stack, which contains the slice. The index pair (m, i) can be uniquely represented by a single index k, computed from (m, i) according to Eq. (3.1).

denote the binary indicator function, which returns 1 if and only if a pixel's intensity value ${}_kI(p) \in \mathbb{R}$ in the k-th slice belongs to the j-th intensity bin for $j \in \{0, \dots, n_{bins} - 1\}$, i.e.

$$\chi_k(p, j) := \begin{cases} 1, & \text{if } {}_kI(p) \in \mathscr{B}_j \\ 0, & \text{else.} \end{cases}$$

Then the normalized intensity histogram

$$h_k : \{0, \dots, n_{bins} - 1\} \to [0, 1]$$

of the femur area within the k-th axial slice is calculated by

$$h_k(j) := \frac{\sum_p \chi_k(p, j) \cdot {}_kGT(p)}{\sum_p {}_kGT(p)}.$$

The feature vectors are defined by means of these intensity histograms, representing discrete intensity distributions, i.e.

$$v_k := (h_k(0), \dots, h_k(n_{bins-1}))^T \in [0, 1]^{n_{bins}} \tag{3.2}$$

for $k = 0, \dots, K - 1$. Therefore, each axial slice of the training set, containing any femur area, yields a feature vector, representing the intensity distribution within the femur area.
Dimensionality reduction by means of Principal Component Analysis (PCA) yields a matrix

$$\mathscr{U} := (u_0, \dots, u_{n_{PCA}-1})$$

containing $n_{PCA} < n_{bins}$ eigenvectors, and a set with the corresponding eigenvalues

$$\mathscr{E} := \{\lambda_0, \dots, \lambda_{n_{PCA}-1}\}.$$

Given the mean intensity distribution

$$\overline{v} := \frac{1}{K} \sum_{k=0}^{K-1} v_k,$$

the first part of the Polar Appearance Models is defined as

$$\mathscr{M} := (\mathscr{U}, \mathscr{E}, \overline{v}),$$

which models the intensity distribution

Modeling border transition in Polar Space

For the second part, the border transition from the femur to its surrounding is analyzed in polar space for all axial training slices. Each axial slice, in which a femur fraction is present, is transformed into polar space, where the centroid of the femur fraction is used as the origin. In the following the x-axis describes the phase and the y-axis the radius in polar space. For each phase position, an intensity profile of length $n_{PC} > 1$ along the y-axis is extracted at the border between the femur boundary and its surrounding. The use of these profiles in polar space is closely related to the Appearance Model of Active Shape Models [CT92] in euclidean space. Let $\tilde{K}$ denote the total number of intensity profiles $\tilde{v}_0, \ldots, \tilde{v}_{\tilde{K}-1}$, acquired in polar space. The application of PCA again yields a matrix

$$\tilde{\mathscr{U}} := (\tilde{u}_0, \ldots, \tilde{u}_{\tilde{n}_{PCA}-1})$$

containing $\tilde{n}_{PCA} < n_{PC}$ eigenvectors, and a set with the corresponding eigenvalues $\tilde{\mathscr{E}} := \{\tilde{\lambda}_0, \ldots, \tilde{\lambda}_{\tilde{n}_{PCA}-1}\}$. Given the mean intensity profile

$$\tilde{\overline{v}} := \frac{1}{\tilde{K}} \sum_{k=0}^{\tilde{K}-1} \tilde{v}_k,$$

the second part of the Polar Appearance Model is defined as

$$\tilde{\mathscr{M}} := (\tilde{\mathscr{U}}, \tilde{\mathscr{E}}, \tilde{\overline{v}}).$$

The final model comprises both parts, i.e.

$$\mathscr{P} := (\mathscr{M}, \tilde{\mathscr{M}}).$$

3.3.2 Center Line Extraction

After training the PAMs as described in section 3.3.1, the subsequent stage deals with the localization of the femur position within unseen MR volumes without using any additional ground truth labels. For this purpose, the PAMs, which have been trained from available training atlases, are utilized. The general idea is to locate the center line of the femur, particularly employing the first intensity distribution model component of the PAMs.
The *center line* is defined as the chain of femur centroids along the axial slices of a volume. Since a convex femur shape in axial slices is assumed, a straight forward approach is the use of the Hough Transform [Hou62] for the detection of circular structures in each slice. The Hough Transform detects probable center points of

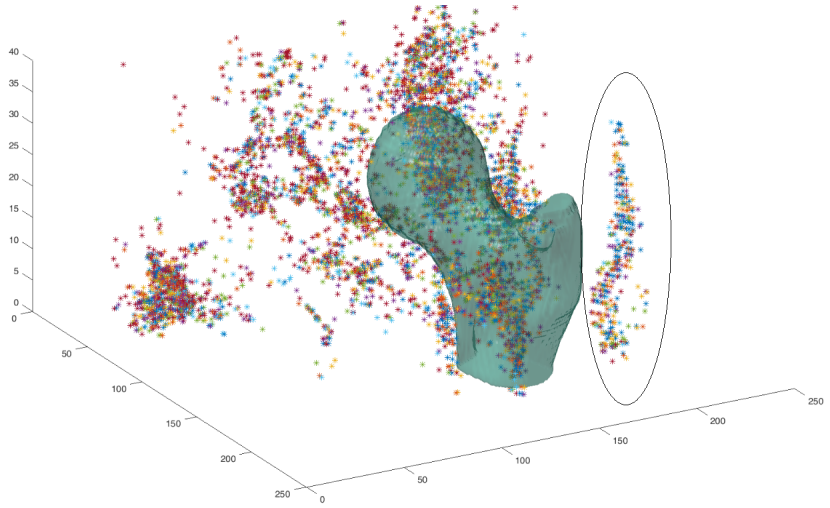


FIGURE 3.3: Hough Transform on axial slices results in a large set of center line point candidates. Some candidates even seem to form a decent path, but are not within the femur, e.g. the marked candidate path on the right. Copyright ©2018 IEEE [Pha+18].

circles and the corresponding radii. These candidate center points of the detected circles can be used as proposals for the center line points.

Establishing a path of center points from bottom axial slice to top axial slice yields an approximated path proposal of the actual femoral center line. A major challenge, however, is the presence of many different candidates in each axial slice and even more candidates in the whole volume. This renders the localization of a feasible center line a difficult task. In Fig. 3.3 the detected center points from the Hough Transforms are visualized. It also demonstrates that there are also feasible center point paths that lie outside the actual femur area.

Because of the sequential nature of the stack of axial slices, the path finding problem can be modeled as a multi-stage optimization problem. To make use of the PAMs' first component $\mathscr{M}$, the intensity distribution within the circular area of each candidate point p, that is detected by the Hough Transform, is represented by a discrete histogram vector $v(p)$ of length n_{bins}, similar to Eq. (3.2). An intensity distribution similarity distance $s(p)$ of $v(p)$ to the learned intensity distributions in $\mathscr{M}$ is defined as the weighted distance of the distribution $v(p)$ to its origin in eigenspace, i.e.

$$s(p) := \sum_{i=0}^{n_{PCA}-1} \frac{1}{\lambda_i} \left({u_i}^T \cdot (v(p) - \overline{v}) \right)^2 . \tag{3.3}$$

Dividing by the corresponding eigenvalues results in punishing deviations in major principal directions less than deviations in minor principal directions. This is equivalent to the squared *Mahalanobis* distance [DMJRM00]. Let $\hat{M}$ denote the total number of axial slices in the unseen MR volume and let P_i denote the set of detected candidate points from the Hough Transform in the i-th axial slice for $i \in \{0, \ldots, \hat{M} - 1\}$. Then $p_i \in P_i$ denotes a candidate center point in slice i and $p_{i-1} \in P_{i-1}$ a candidate point in slice $i - 1$. To find a suitable path as center line approximation, a cost is assigned to each connection between candidate points in adjacent axial slices. The cost function is denoted as $c(p_{i-1}, p_i)$ and may comprise several distinguishing aspects,

which are further discussed in section 3.3.2. The center line detection is reduced to the solution of the multi-stage optimization problem

$$\underset{p_0,\ldots,p_{\hat{M}-1}}{\operatorname{argmin}} \sum_{i=1}^{\hat{M}-1} c(p_i, p_{i-1}) + w_s s(p_i), \tag{3.4}$$

where $p_i \in P_i$ for $i \in \{0, \ldots, \hat{M}-1\}$, and $w_s > 0$ is a weighting factor for the similarity distance (Eq. (3.3)). Finding a suitable path can therefore be achieved by means of dynamic programming. To address the optimization problem, a sequence of cost functions

$$F_i : P_i \to \mathbb{R}$$

measuring the minimal path's cost until the i-th axial slice for any candidate point $p_i \in P_i$ is defined recursively by

$$\begin{aligned} F_0(p_0) &:= 0 \qquad \forall p_0 \in P_0 \\ F_{i+1}(p_{i+1}) &:= \underset{p_i \in P_i}{\operatorname{argmin}} \{F_i(p_i) + c(p_{i+1}, p_i) + w_s s(p_{i+1})\}. \end{aligned} \tag{3.5}$$

By means of iterative dynamic programming, the path leading to the candidate point $p_{\hat{M}-1} \in P_{\hat{M}-1}$ with least cost according to $F_{\hat{M}-1}$ is selected as estimated femur center line. Although there are some non-circular sections within the femur, the shaft and head slices mostly yield near-circular bone areas, such that the proposed optimization procedure compensates possible erroneous center point candidates in the corresponding slices e.g. by penalizing large distances between adjacent center points. To restrict the length of the center line, specifically for the femoral model initialization, positions in which a rapid change of center line radius occurs, can be used to terminate the path.

Cost Function Design

The specific formulation of the cost function $c(p_{i-1}, p_i)$ for the center line localization is highly dependent on the application and therefore a design choice. In the context of femur extraction in MR volumes, the following weighted aspects are considered:

- **Center Point Shift**
  Since a path is expected with only small center point shifts between adjacent slices, any shift in center point proposals is punished by it's $L2$-norm, i.e.

  $$w_d ||p_{i-1} - p_i||_2,$$

  where $w_d > 0$ denotes the weighting factor of this component.

- **Mean Intensity Shift**
  A further indicator of whether the connection of the adjacent candidate points p_{i-1} and p_i may be feasible is the deviation of mean intensities within their circular area. It is expected, that there is only a small shift in mean intensity, thus large deviations may also be punished. Let $\mu(p_{i-1})$ and $\mu(p_i)$ denote the mean intensities within the circular area of the corresponding candidate points. Then the weighted absolute difference

  $$w_\mu |\mu(p_{i-1}) - \mu(p_i)|$$

serves as an additional cost term, with $w_\mu > 0$.

- **Variance Shift**
  Similarly the deviation in intensity variance within the circular areas can be used to penalize their connection, i.e.

  $$w_{var}|var(p_{i-1}) - var(p_i)|$$

  for $w_{var} > 0$ and with $var(\cdot)$ referring to the intensity variance within the circular area.

- **Radius Shift**
  Although the axial femur cross section changes from one slice to another, it can be assumed that the cross section area does not undergo rapid changes. Therefore strong radius deviations from one slice to the next of the circular candidate points can also be used to determine the suitability of two adjacent candidate points, i.e.

  $$w_r|r(p_{i-1}) - r(p_i)|$$

  where $w_r > 0$ serves as weighting factor and $r(\cdot)$ represents the radius of a candidate point, proposed by the Hough Transform.

- **Accumulated Mean Intensity Shift**
  Instead of measuring the mean intensity difference of circular areas of only adjacent candidate points p_{i-1} and p_i, an alternative approach is the calculation of the deviation of the next candidate point's mean intensity μ_{p_i} from the accumulated mean intensity of all circular areas of already selected previous candidate points $p_0, \ldots, p_{i-1}$, which will be abbreviated as $\tilde{\mu}(p_{i-1})$, i.e.

  $$w_{\tilde{\mu}}|\tilde{\mu}(p_{i-1}) - \mu(p_i)|$$

  for $w_{\tilde{\mu}} > 0$.

- **Accumulated Center Point Shift**
  With the same argumentation, the deviation of the next candidate point's center position p_i from the previously selected positions $p_0, \ldots, p_{i-1}$, denoted as $\overline{p_{i-1}}$, can be assessed by

  $$w_{\tilde{d}}||\overline{p_{i-1}} - p_i||_2$$

  with weight factor $w_{\tilde{d}}$.

- **Center Point Shift from first slice**
  The last component of the cost function, considered in the scope of this work, is the center point shift from the first axial center point selection, i.e.

  $$w_{d_0}||p_0 - p_i||_2$$

  with $w_{d_0} > 0$. This idea is motivated by the assumption, that the femur shaft lies almost perpendicular to the axial plane.

With these components in mind, a weighted cost function, determining the feasibility of connecting candidate points p_{i-1} and p_i from adjacent axial slices, can be

formulated by

$$\begin{aligned} c(p_{i-1}, p_i) := \; & w_d||p_{i-1} - p_i||_2 \\ & + w_\mu|\mu(p_{i-1}) - \mu(p_i)| \\ & + w_{var}|var(p_{i-1}) - var(p_i)| \\ & + w_r|r(p_{i-1}) - r(p_i)| \\ & + w_{\tilde{\mu}}|\tilde{\mu}(p_{i-1}) - \mu(p_i)| \\ & + w_{\tilde{d}}||\tilde{d}(p_{i-1}) - p_i||_2 \\ & + w_{d_0}||p_0 - p_i||_2. \end{aligned} \tag{3.6}$$

A more detailed hyper parameter analysis for the weight selection and an investigation of the impact of the proposed cost function components is presented in section 3.3.5

3.3.3 Boundary Detection in Polar Space

In the third stage, the extracted femur center line and the corresponding radii from the center points are used to apply a transformation of the axial slices from Euclidean space into polar space. Given the center line and the corresponding radii of the Hough Transform from the previous stage, it is possible to restrict the search area for the femur in each axial slice to a circular area around the center line with a radius approximately as large as the largest center line radius. Each axial MRI slice can be transformed into a restricted polar space with the corresponding center line point as origin (see Fig. 3.4 (a)).

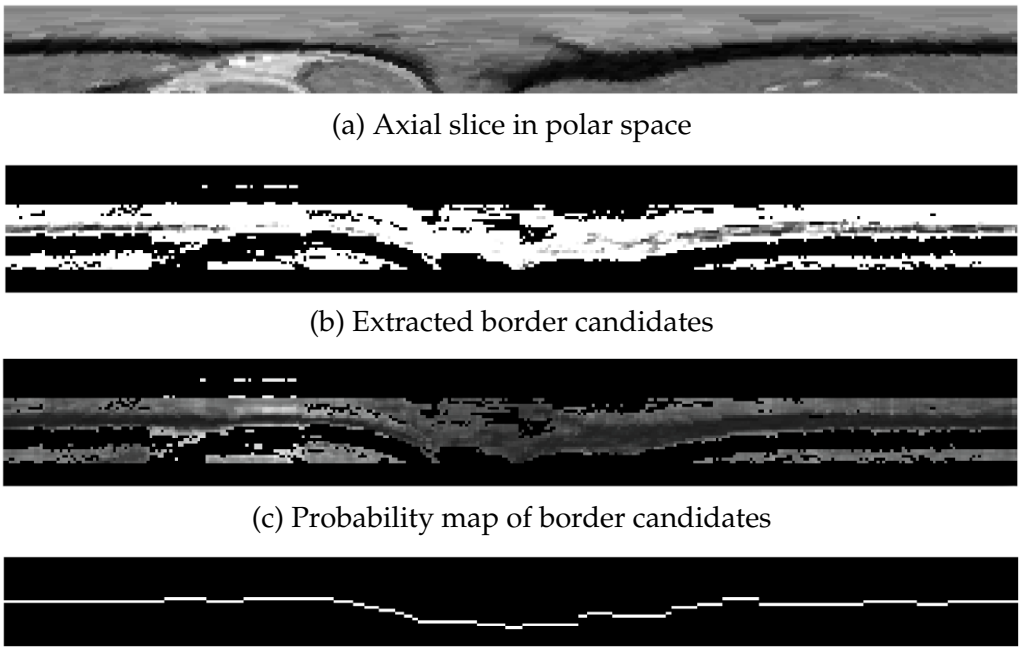
For each column, i.e. phase, multiple profile vectors of length n_{PC} along the y-axis, i.e. the amplitude, can be generated, where the center position of these vectors yield possible border point candidates. Let $H_{ps} \times W_{ps}$ denote the size of the polar transformed axial slice. Then for each column $H_{ps} - n_{PC}$ candidate border points with corresponding profile vectors can be determined. Let $\tilde{v}(p)$ denote the intensity profile vector of a candidate boundary point p, where p is located in the center position of the vector $\tilde{v}(p)$. The similarity of each candidate profile vector to the learned profile $\tilde{s}(p)$ can be measured in Eigenspace using $\tilde{\mathscr{M}}$ in the same way as described in Eq. (3.3), i.e.

$$\tilde{s}(p) := \sum_{i=0}^{\tilde{n}_{PCA}-1} \frac{1}{\tilde{\lambda}_i} \left(\tilde{u}_i^T \cdot (\tilde{v}(p) - \bar{\tilde{v}})\right)^2. \tag{3.7}$$

In each column the most likely borderline candidates are distinguished by means of Eq. (3.7), as can be seen in Fig. 3.4 (b). Fig. 3.4 (c) shows the probability map of the selected borderline points of (b) according to the similarity measure.

Let $\tilde{P}_i$ denote the set of possible boundary candidate points in the i-th column for $i \in \{0, \ldots, W_{ps} - 1\}$. Then the problem of finding the femur boundary can be reduced to finding a path of boundary points from $\tilde{P}_0$ to $\tilde{P}_{W_{ps}-1}$. Assigning a cost to each connection between boundary candidate points $p_{i-1} \in \tilde{P}_{i-1}$ and $p_i \in \tilde{P}_i$ in adjacent columns for $i \in \{1, \ldots, W_{ps} - 1\}$, aids in establishing a suitable boundary path. The cost function is denoted as $\tilde{c}(p_{i-1}, p_i)$ and is defined as

$$\tilde{c}(p_{i-1}, p_i) := ||p_{i-1} - p_i||_2,$$



(a) Axial slice in polar space

(b) Extracted border candidates

(c) Probability map of border candidates

(d) Resulting borderline

FIGURE 3.4: Exemplary borderline extraction in polar space.
Copyright ©2018 IEEE [Pha+18].

only taking the Euclidean distance of adjacent boundary point candidates into account. With the same methodological approach as in Eq. (3.4), the boundary approximation is reduced to the solution of the multi-stage optimization problem

$$\underset{p_0,\dots,p_{W_{ps}-1}}{\operatorname{argmin}} \sum_{i=1}^{W_{ps}-1} \tilde{c}(p_i, p_{i-1}) + I(p_i)\tilde{s}(p_i), \tag{3.8}$$

where $p_i \in \tilde{P}_i$ for $i \in \{0, \dots, W_{ps} - 1\}$. A major difference in Eq. (3.8) to Eq. (3.4) is that the similarity distance $\tilde{s}(p)$ is weighted with the intensity $I(p)$ at point p. This is motivated by the fact, that the borderline is ordinarily located in dark regions of the MR volume.

Formulating a recursion as in Eq. (3.5) for dynamic programming

$$\begin{aligned} \tilde{F}_0(p_0) &:= 0 \qquad \forall p_0 \in \tilde{P}_0 \\ \tilde{F}_{i+1}(p_{i+1}) &:= \underset{p_i \in \tilde{P}_i}{\operatorname{argmin}} \{\tilde{F}_i(p_i) + \tilde{c}(p_{i+1}, p_i) + I(p_{i+1})\tilde{s}(p_{i+1})\}. \end{aligned} \tag{3.9}$$

leads to an optimal path from the left to the right column, representing the approximated borderline in polar space. Transforming these approximations back to euclidean space yields a coarse estimation of the femur's contour (see Fig. 3.4 (d)). This coarse estimation is finally used to apply an ICP approach to calculate a transformation of an arbitrary shape model to fit into the point cloud consisting of the border estimates.

3.3.4 Model Fitting

The last stage consists of fitting an arbitrary shape model into the boundary point cloud. Rigid ICP [BM92] is particularly utilized to compute a rough transformation estimate, that is applied on the shape model to achieve a reasonable initialization point for a subsequent segmentation method. To speed up the registration process, only a subset of the shape model's boundary points is used for ICP by means of equidistant subsampling.

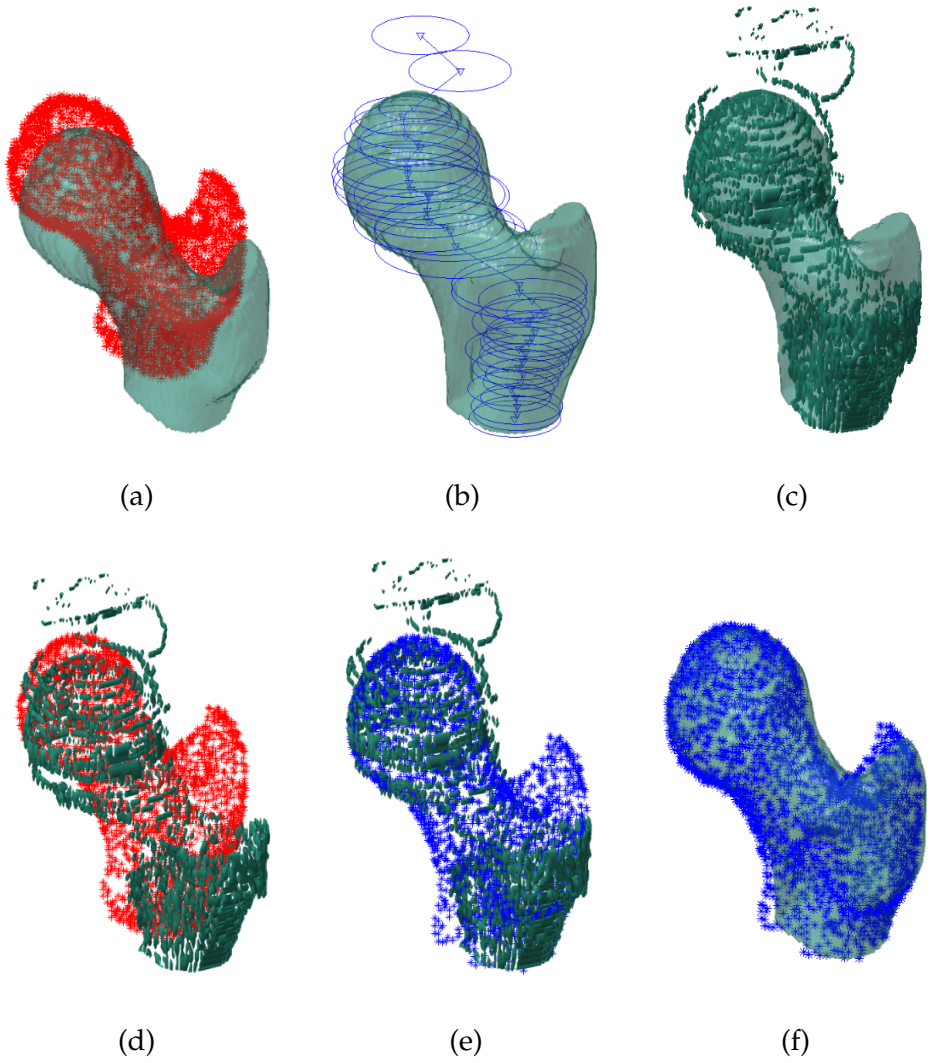


FIGURE 3.5: (a) Overlay of unregistered shape model (red) and ground truth (green). (b) Extracted center line from unseen volume during stage 2. (c) Resulting femur border from stage 3. (d) Original displacement of shape model (red) and femur border. (e) Fitted shape model (blue) to femur border. (f) Overlay of transformed shape model (blue) and ground truth (green). Copyright ©2018 IEEE [Pha+18].

Fig. 3.5 shows exemplary snapshots of the proposed approach. To illustrate the qualitative results of the procedure in each snapshot, the desired ground truth segmentation is overlayed in green. It should be noted, that the displayed ground truth of the unseen MR volume only serves as visual inspection and is not used for the initialization process, neither during the training stage nor during the remaining

stages. Fig. 3.5 (a) shows the initial position of the shape model (red) and its deviation from a more suitable positioning close to the depicted ground truth (green). The approximated femur center line with corresponding circle estimates from the second stage is shown in Fig. 3.5 (b). It is observable that the proposed center line has a feasible progression through the femur. Fig. 3.5 (c) shows the achieved boundary points from the third stage. In Fig. 3.5 (d) the initial displacement of shape model (red) and estimated boundary points is shown, whereas the registered shape model (blue) via ICP during stage four is depicted in Fig. 3.5 (e). Fig. 3.5 (f) shows the transformed shape model (blue) with an overlay of the ground truth (green). It is noticeable that the transformed shape model serves as a better initialization for subsequent segmentation methods than the original positioning in Fig. 3.5 (a).

3.3.5 Experiments

The following paragraph describes the experiments conducted with the proposed PAMs approach for femoral model initialization. First, the data sets that are used for the experiments are presented. There are several aspects that can be analyzed regarding the proposed initialization method. The subsequent segment presents the achieved initialization results in a leave-one-out cross validation setting. Because of the 2D nature in the training stage, it is feasible to assume that one training volume may be already sufficient for a satisfactory initialization. Therefore, the feasibility of using only one training volume is additionally investigated in the subparagraph following the next one. The last subsection particularly considers the aspect of retrospective hyper parameter analysis regarding the cost function (Eq. (3.6)), described in section 3.3.2.

Data

For the experiments, eight T1-weighted MR volumes of the femur from six different patients are utilized. Two of these patients were examined before and after surgical procedures. The MR images were recorded using a *Siemens Magnetom Aera 1,5 Tesla* MR tomograph during clinical routine and are provided by the Department of Orthopaedics and Trauma Surgery at the University Hospital Essen. The patient volumes are denoted as P1, . . . ,P6 and the post operative data sets are marked as P1PO and P2PO, respectively. Table 3.1 shows the voxel spacing and the volume size of the considered patient volumes.

Patient Volume	Voxel Spacing	Volume Size
P1	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 40)$
P1PO	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 48)$
P2	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 40)$
P2PO	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 48)$
P3	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 40)$
P4	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 40)$
P5	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 48)$
P6	$(0.89 \times 0.89 \times 2.4)$	$(256 \times 256 \times 48)$

TABLE 3.1: Resolution of femur MR data sets.

Leave-One-Out Model Initialization

In a leave-one-out cross validation manner, each available patient volume P1, . . . ,P6 and P1PO, P2PO is considered as test volume once, whereas the remaining patient volumes and their corresponding label volumes are considered training volumes and used to form the PAMs in the training stage. It should be noted that only MR volumes are used for training, which are not of the same patient as the test volume, e.g. P1 was not used during training for the test volume P1PO. A multi-atlas 3D non-rigid diffeomorphic demons registration [Ver+09] implementation in MATLAB is used as a reference baseline (Multi-Atlas I). Additionally, *Elastix* is used for a second multi-atlas approach with the parameter settings of Bron et al. [Bro+13] for knee cartilage registration in MRI (Multi-Atlas II). The parameter settings are taken from the Elastix Model Zoo[1].

w_s	w_d	w_μ	w_{var}	w_r	$w_{\tilde{\mu}}$	$w_{\tilde{d}}$	w_{d_0}
10	5	0	0	1	0	0	0

TABLE 3.2: Weight configuration for experiments.

Table 3.2 shows the weight configuration, chosen for the center line extraction in stage 2. These values focus on a high weighting of the similarity component. A more detailed assessment on the hyper parameter selection is presented in section 3.3.5. For the last model fitting stage, the label image of P2PO is arbitrarily used to be registered to the extracted boundary points. For the folds, in which P2 and P2PO are the test volume, P1's label volume is used as shape model to be registered.

DSC	Multi-Atlas I	Multi-Atlas II	PAMs
P1	0.3082	0.6919	0.8646
P1PO	0.2925	0.4974	0.8170
P2	0.2851	0.7373	0.7031
P2PO	0.2613	0.6586	0.6602
P3	0.2418	0.5840	0.8242
P4	0.1229	0.1562	0.7999
P5	0.1109	0.6161	0.8081
P6	0.2729	0.3912	0.9132
$\emptyset$	0.2370 ± 0.0768	0.5416 ± 0.1450	0.7988 ± 0.0586

TABLE 3.3: Resulting DSCs from the proposed PAMs approach compared to the multi-atlas baselines.

Table 3.3 shows the achieved initialization results of both multi-atlas registration approaches for each testing fold compared to the proposed PAMs pipeline. It is noticeable that for most patient volumes the multi-atlas approaches performs poorly with mean dice scores of 0.2370 and 0.5416 compared to the proposed PAMs' approach achieving 0.7988 on average. The multi-atlas methods' poor performance may be due to the different field of views (FOVs) of the MRI data sets. The inconsistency of the FOV increases the difficulty of the registration problem, as a larger FOV shows more anatomical structures, which cannot be matched to images with smaller FOVs. This is circumvented in the PAMs approach, in which the registration is not applied

[1] https://elastix.lumc.nl/modelzoo/par0017/

on gray scale volumes, but on point clouds of the particular structure of interest, which is the proximal femur in this case. The different FOVs of the data set are illustrated in Fig. 3.6, where the ground truths of arbitrary volumes with very different FOVs are shown in comparison.

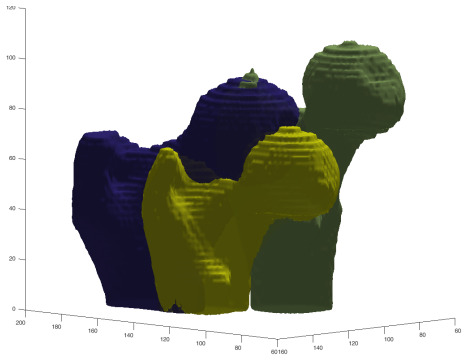


FIGURE 3.6: Exemplary ground truths of volumes with different FOVs compared.

Model Initialization with one Training Volume

As the presented PAMs pipeline leverages the 2D sequential nature of MR volumes, it has access to a significantly increased number of training samples for the training stage compared to the mere number of patients (see section 3.3.1). Therefore, in this subparagraph the feasibility of using only one patient training volume, i.e. a single patient training approach, for a satisfactory femur model initialization is investigated.

Test \ Atlas	P1	P1PO	P2	P2PO	P3	P4	P5	P6	∅	std
P1	-	0.946	0.586	0.585	0.447	0.007	0.802	0.209	0.512	± 0.249
P1PO	0.943	-	0.361	0.638	0.311	0.006	0.579	0.684	0.503	± 0.237
P2	0.233	0.051	-	0.939	0.735	0.381	0.333	0.322	0.428	± 0.234
P2PO	0.494	0.275	0.904	-	0.657	0.494	0.388	0.535	0.535	± 0.140
P3	0.153	0.069	0.816	0.541	-	0.253	0.406	0.250	0.356	± 0.199
P4	0.107	0.100	0.626	0.522	0.731	-	0.733	0.576	0.485	± 0.218
P5	0.726	0.507	0.364	0.372	0.262	0.160	-	0.523	0.416	± 0.145
P6	0.368	0.311	0.464	0.802	0.352	0.475	0.623	-	0.485	± 0.130
∅	0.432	0.323	0.589	0.628	0.499	0.254	0.552	0.443	0.465	± 0.204
std	± 0.248	± 0.231	± 0.166	± 0.141	± 0.179	± 0.168	± 0.151	± 0.156		

TABLE 3.4: Resulting DSCs from *Elastix* approach with single patient training. Columns denote the atlas data set, rows depict the test data set.

For this, each train-test combination of the available volumes is assessed, such that only one volume is used for training in the first stage, and its corresponding label image is employed as shape model, which is registered to the extracted boundary points in the final fitting stage. For the center line extraction stage, the same weight configuration, depicted in Table 3.2 is used for all experiments.
Instead of a multi-atlas registration, a simple atlas registration by means of the *Elastix* configuration of section 3.3.5 is applied for every possible train-test combination to form a baseline. This deviation from the original publication [Pha+18] is presented, since the *Elastix* implementation yields superior results compared to the originally

used registration. The achieved dice scores of the registration approach for each combination is shown in Table 3.4, whereas the resulting dice scores for the PAMs pipeline is presented in Table 3.5.

Test \ Train	P1	P1PO	P2	P2PO	P3	P4	P5	P6	∅	std
P1	-	0.899	0.878	0.884	0.662	0.723	0.690	0.705	0.777	±0.094
P1PO	0.888	-	0.783	0.847	0.680	0.670	0.316	0.158	0.620	±0.192
P2	0.859	0.841	-	0.910	0.817	0.583	0.843	0.897	0.821	±0.061
P2PO	0.836	0.797	0.897	-	0.509	0.509	0.521	0.000	0.581	±0.197
P3	0.815	0.816	0.850	0.841	-	0.888	0.476	0.831	0.788	±0.078
P4	0.572	0.782	0.854	0.833	0.888	-	0.001	0.839	0.681	±0.197
P5	0.771	0.764	0.058	0.817	0.733	0.747	-	0.822	0.673	±0.154
P6	0.826	0.862	0.880	0.758	0.786	0.821	0.814	-	0.821	±0.026
∅	0.795	0.823	0.743	0.841	0.725	0.706	0.523	0.607	0.720	± 0.158
std	±0.071	±0.038	±0.196	±0.033	± 0.093	± 0.102	±0.222	± 0.302		

TABLE 3.5: Resulting DSCs from the proposed PAMs approach with single patient training. Columns denotes the training data set, rows depict the test data set. Copyright ©2018 IEEE [Pha+18].

The registration baseline achieves a mean Dice Similarity Coefficient (DSC) of 0.465 with a standard deviation of ± 0.204, whereas the PAMs approach results in a significantly improved average dice score of 0.720 with a standard deviation of ± 0.158. A limitation of the PAMs approach is that the automatic detection of the center line is crucial to the final initialization, since the subsequent steps do not offer any correction mechanism. This becomes especially imminent in very weak DSC cases, such as the combination of P6 and P2PO, where the DSC is zero. In this case the wrong center line is extracted, which is propagated to the subsequent stages, rendering a poor performance. However, it needs to be emphasized, that only one training volume is used to train the PAM model and that a larger training set, similar to a multi-atlas approach, would result in an even more robust center line extraction, as the leave-one-out experiments in section 3.3.5 demonstrate.

Retrospective Hyper Parameter Analysis

This section takes a closer look at the cost function (Eq. (3.6)) for the center line estimation (stage 2), proposed in section 3.3.2. The aim of this paragraph is to investigate the impact of the individual components of the cost function and the necessity of the similarity component (Eq. (3.3)). Therefore, this subsection also takes a look at the feasibility of the PAMs' training stage. For this purpose a grid search for the best weight configuration is applied, where the achieved DSC after initialization is used as a quantitative metric. To restrict the search space, only discrete weight configurations are considered, which sum up to eight (since there are 8 components to be assessed). The weights are only taken from a grid of natural numbers ranging from $\{0, \ldots, 8\}$. Using a leave-one-out cross validation strategy, as described in the previous section 3.3.5, the mean DSC over all test volumes is calculated for every allowed weight combination.

The results are illustrated in Fig. 3.7 by means of a color encoded parallel plot. The proposed cost function components and the similarity function (Mahalanobis Distance) are denoted on the x-axis, whereas their weights are represented on the y-axis. The last two pillars on the right denote the achieved mean DSC, visualizing their color encoding. Each path from left to right represents one of the considered weight

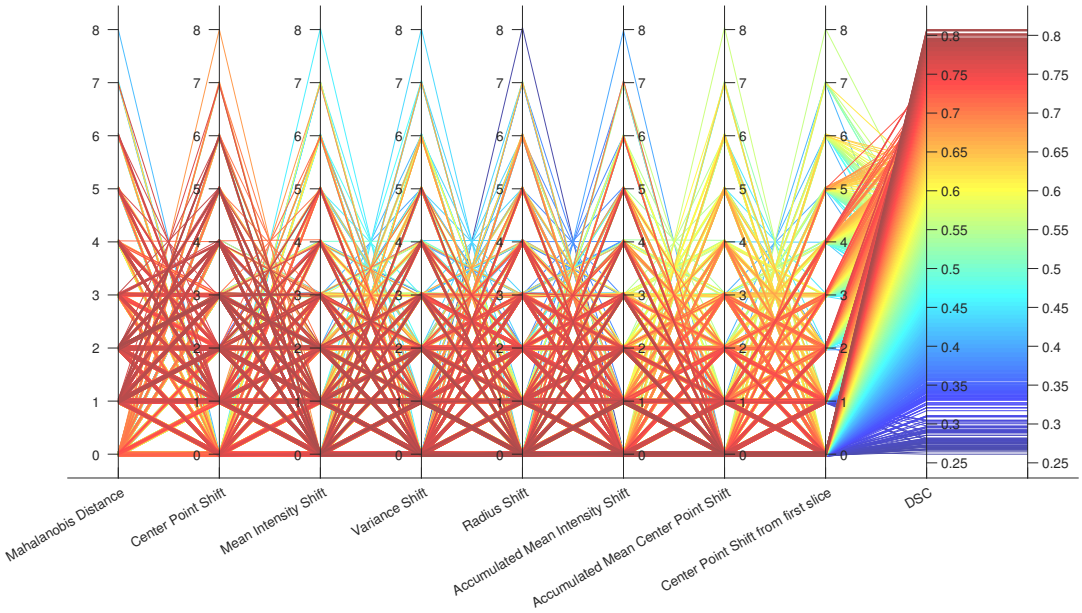


FIGURE 3.7: PAMs parallel plot.

combinations, where a color close to red indicates a good performance, while a color close to blue represents a weak initialization result. Therefore, those cost function components with pillars, which show rather blue than red incoming and outgoing edges at the top, indicate a lower importance for a successful initialization, since a high weighting only results in poor performance.
It is observable, that the center point shift and the similarity component appear to be most relevant according to the color encoding at their pillar's top region. In contrast, the aspects of accumulated mean center point shift and center point shift from the first axial slice seem to be neglectable. Furthermore, a more well-balanced weight combination often results in favorable mean DSCs, as the red appearance of the lower weight regions suggests.

	DSC	w_s	w_d	w_μ	w_{var}	w_r	$w_{\tilde{\mu}}$	$w_{\tilde{d}}$	w_{d_0}
P1	0.8966	2	4	0	0	1	1	0	0
P1PO	0.9007	2	1	0	0	5	0	0	0
P2	0.7294	0	7	1	0	0	0	0	0
P2PO	0.6928	2	0	1	0	3	2	0	0
P3	0.8606	1	1	0	4	2	0	0	0
P4	0.8475	3	4	0	0	0	0	0	1
P5	0.8300	6	1	0	0	0	0	0	1
P6	0.9194	1	4	0	2	0	0	0	1
$\emptyset$	0.8346	2.125	2.75	0.25	0.75	1.375	0.375	0	0.375
Best Overall	0.8074	2	5	0	1	0	0	0	0

TABLE 3.6: Best weight configuration for each patient. $\emptyset$ denotes the mean over all patients. *Best Overall* denotes the specific configuration, resulting in the best *mean* DSC.

Tab. 3.6 shows the best parameter configuration for each patient. It is noticeable, that the accumulated mean center point shift, weighted by $w_{\tilde{d}}$, does not contribute to the best parameter configuration for any test patient. Furthermore, the weights

are averaged along all patients for a rough estimate of their importance (second to last row). Here, the previous observation from Fig. 3.7 is supported, as the center point shift and the similarity component are weighted more heavily on average than the remaining components. The best mean DSC of 0.8074, considering all test patients, is also achieved by strong weights on the center point shift and the similarity component, as can be seen in the last row. In summary, a heavy emphasis on center point shift and similarity component in the cost function is preferable, although the weighting ratio of these two cost function components to each other is neglectable, which is reflected in the similarly high mean DSC result of 0.7988 using the original preliminary weight configuration of Tab. 3.2, which has a strong focus on both center point shift and the similarity component, but weights the similarity component more heavily.

3.4 GESAM: Gradient based Expanding Spherical Appearance Models

The second initialization approach of this chapter is a three stage localization method, as visualized in Fig. 3.8, that finds a feasible contour initialization for a subsequent contour based segmentation approach. A prerequisite is the presence of a near-spherical component within the anatomical structure, that needs to be extracted. In case of the proximal femur, its femur head represents said near spherical component. The first stage consists of training the Gradient based Expanding Spherical Appearance Models (GESAMs) to model the intensity distribution within the femur head and its local environment in a Principal Component Analysis (PCA) manner. In the second stage, the trained model is used to robustly localize the near-spherical component, i.e. the femur head, of the proximal femur. It consists of a preprocessing, a structured sampling, and a sphere selection step. In the final stage, the estimated femur head location is used as an initial region to expand to the remaining bone regions by a simplified Level Set approach, which is restricted by the MR volume's gradient information.

3.4.1 Model Definition

Like in the previous section, let

$$(I_0, GT_0), \ldots, (I_{N-1}, GT_{N-1})$$

denote N training atlases, with MR volume I_i and its corresponding label volume GT_i for $i = 0, \ldots, N-1$. It is assumed that each volume has an isotropic voxel spacing of $(1 \times 1 \times 1)mm^3$. During the first stage, the GESAMs learn to model the intensity distribution within and around the near-spherical component. For this reason, the near-spherical components need to be identified within the label volumes. This is achieved by fitting a sphere by Random Sampling Consensus (RANSAC) [FB81] into the label volume of each atlas. Let r_i be the radius of the fitted sphere in GT_i for $i = 0, \ldots, N-1$, then an encapsulating outer neighborhood of the sphere can be extracted, such that the volume of the outer neighborhood is equal to the sphere volume. This is achieved by defining the neighborhood bandwidth of

$$bw_i := \sqrt[3]{2} r_i - r_i. \tag{3.10}$$

Fig. 3.9 shows an example of the detected femur head (red) and its outer neighborhood (gray). After the sphere fitting procedure, an inner and outer region with the same volume can be therefore differentiated. For a fixed number $n_{bins} \in \mathbb{N}\backslash\{0\}$ of intensity bins, the normalized intensity distributions of the inner and outer regions are modeled by means of two vectors $v_i^{in}, v_i^{out} \in [0,1]^{n_{bins}}$, yielding N feature vectors $v_0^{in}, \ldots, v_{N-1}^{in} \in [0,1]^{n_{bins}}$ for the inner region, and N feature vectors $v_0^{out}, \ldots, v_{N-1}^{out} \in [0,1]^{n_{bins}}$ for the outer region.
For the calculation of the normalized intensity distribution, the whole intensity range is divided into the aforementioned n_{bins} intensity bins, i.e. $\mathscr{B}_0, \ldots, \mathscr{B}_{n_{bins}-1}$. If e.g. the maximally possible intensity value is 255 and the minimally possible value is 0, then the bins would be defined as

$$\mathscr{B}_0 := [0, 128)$$
$$\mathscr{B}_1 := [128, 255]$$

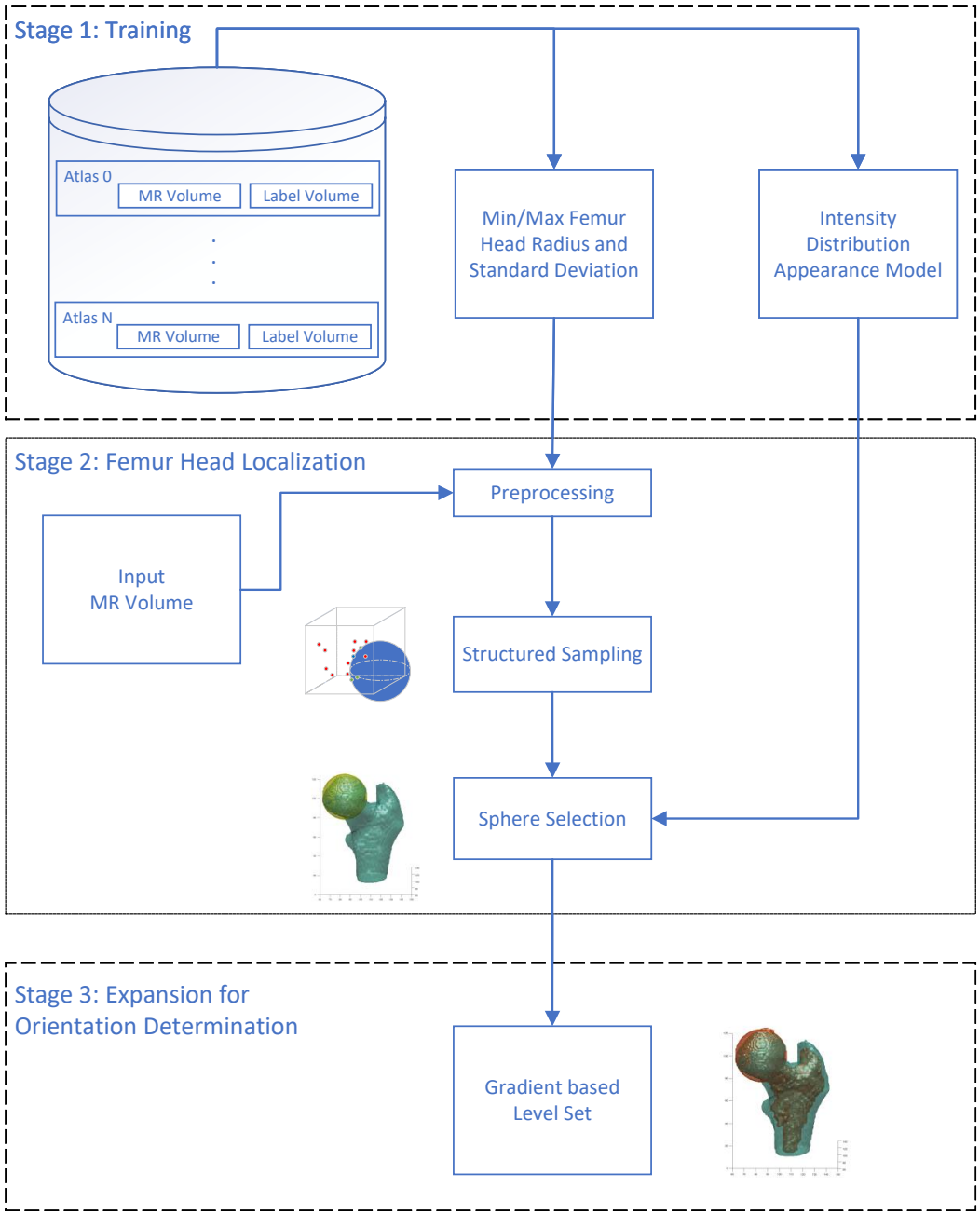


FIGURE 3.8: Overview of GESAMs strategy.

for $n_{bins} := 2$. Let

$$\chi_i : \Omega \times \{0, n_{bins} - 1\} \to \{0, 1\} \tag{3.11}$$

denote the binary indicator function, which returns 1 if and only if a voxel's intensity value $I_i(p) \in \mathbb{R}$ in the i-th MR volume belongs to the j-th intensity bin for $j \in$

$\{0, \ldots, n_{bins} - 1\}$, i.e.

$$\chi_i(p, j) := \begin{cases} 1, & \text{if } I_i(p) \in \mathscr{B}_j \\ 0, & \text{else.} \end{cases} \tag{3.12}$$

Then the normalized intensity histogram

$$h_i^{in} : \{0, \ldots, n_{bins} - 1\} \to [0, 1] \tag{3.13}$$

of the inner sphere region within the i-th volume is calculated by

$$h_i^{in}(j) := \frac{\sum_p \chi_i(p, j) \cdot S_i^{in}(p)}{\sum_p S_i^{in}(p)}, \tag{3.14}$$

where $S_i^{in} : \Omega \to \{0, 1\}$ denotes the indicator function of whether $p \in \Omega$ is within the inner region of the fitted sphere in the i-th label volume. In the same fashion, the normalized intensity histogram

$$h_i^{out} : \{0, \ldots, n_{bins} - 1\} \to [0, 1] \tag{3.15}$$

of the outside region surrounding the fitted sphere within the i-th volume is established by

$$h_i^{out}(j) := \frac{\sum_p \chi_i(p, j) \cdot S_i^{out}(p)}{\sum_p S_i^{out}(p)}, \tag{3.16}$$

where $S_i^{out} : \Omega \to \{0, 1\}$ now represents the indicator function if $p \in \Omega$ is within the outer region of the fitted sphere in the i-th label volume.
The feature vectors are defined by means of these intensity histograms, representing discretized intensity distributions, i.e.

$$v_i^{in} := (h_i^{in}(0), \ldots, h_i^{in}(n_{bins-1}))^T \in [0, 1]^{n_{bins}} \tag{3.17}$$

and

$$v_i^{out} := (h_i^{out}(0), \ldots, h_i^{out}(n_{bins-1}))^T \in [0, 1]^{n_{bins}}, \tag{3.18}$$

for $i = 0, \ldots, N - 1$. A third type of feature vector can be constructed by concatenat-

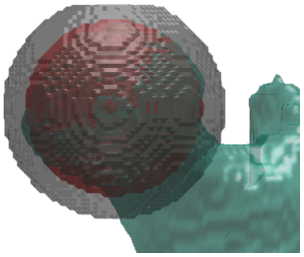


FIGURE 3.9: Visualization of inner and outer region. The red sphere is returned by a sphere detection algorithm (such as RANSAC) on the ground truth data and approximates the femur head. The grey spheres's volume is twice as large as the red one's. The intensity distribution within the outer neighborhood area between these spheres is used to construct a feature vector.

ing these vectors to a joint feature vector of length $2n_{bins}$, i.e.

$$v_i^{joint} := [v_i^{in}, v_i^{out}], \tag{3.19}$$

where $[\cdot, \cdot]$ is the concatenation operator. To reduce the effects of global intensity changes, the derivatives of the distributions are considered. Dimensionality reduction by means of Principal Component Analysis (PCA) yields a matrix for each feature vector type

$$\mathcal{U}^{in} := (u_0^{in}, \ldots, u_{n_{PCA}-1}^{in}) \tag{3.20}$$

$$\mathcal{U}^{out} := (u_0^{out}, \ldots, u_{n_{PCA}-1}^{out}) \tag{3.21}$$

$$\mathcal{U}^{joint} := (u_0^{joint}, \ldots, u_{n_{PCA}-1}^{joint}) \tag{3.22}$$

containing $n_{PCA} < n_{bins}$ eigenvectors, respectively. For each variant a set of eigenvalues can be established by

$$\begin{aligned}
\mathcal{E}^{in} &:= \{\lambda_0^{in}, \ldots, \lambda_{n_{PCA}-1}^{in}\} \\
\mathcal{E}^{out} &:= \{\lambda_0^{out}, \ldots, \lambda_{n_{PCA}-1}^{out}\} \\
\mathcal{E}^{joint} &:= \{\lambda_0^{joint}, \ldots, \lambda_{n_{PCA}-1}^{joint}\}.
\end{aligned}$$

With the mean feature vectors of the aforementioned types

$$\begin{aligned}
\overline{v}^{in} &:= \frac{1}{N} \sum_{i=0}^{N-1} v_i^{in}, \\
\overline{v}^{out} &:= \frac{1}{N} \sum_{i=0}^{N-1} v_i^{out}, \\
\overline{v}^{joint} &:= \frac{1}{N} \sum_{i=0}^{N-1} v_i^{joint}
\end{aligned}$$

a Spherical Appearance Model is defined for each feature variant. i.e.

$$\begin{aligned}
\mathscr{M}^{in} &:= (\mathcal{U}^{in}, \mathcal{E}^{in}, \overline{v}^{in}) \\
\mathscr{M}^{out} &:= (\mathcal{U}^{out}, \mathcal{E}^{out}, \overline{v}^{out}) \\
\mathscr{M}^{joint} &:= (\mathcal{U}^{joint}, \mathcal{E}^{joint}, \overline{v}^{joint}).
\end{aligned}$$

Additionally taking the maximal and minimal radius r_{max}, r_{min} in mm of the fitted spheres, and the standard deviation σ_r of the sphere radii into account, the complete GESAMs model is defined as

$$\mathcal{G} := (\mathscr{M}^{in}, \mathscr{M}^{out}, \mathscr{M}^{joint}, r_{max}, r_{min}, \sigma_r).$$

Therefore, the training stage consists of computing the spherical appearance models and the radius information from the training atlases.

3.4.2 Localization

In the next stage, the GESAMs model $\mathcal{G}$ is used to localize the spherical component of the femur, i.e. the femur head, within a new unseen MR volume. The general idea is to leverage the modeled appearance of the atlases' femur heads and their

surroundings to find similar spherical regions within the volume. This localization stage can be subdivided into three steps, namely

- Preprocessing
- Structured Sampling
- Best Sphere Selection

In the preprocessing step, the unseen MR volume is reduced to a set of feasible boundary points of the femur head. The subsequent structured sampling step is a RANSAC like fitting scheme, i.e. a hypothesis-verification strategy, in which the number of samples, which generate the hypothesis, is kept minimal. In this case the hypothesis is a distinct sphere candidate, specified by a minimal sample set of four voxel coordinates. In the last sphere selection step, the best sphere candidate out of all candidates from the sampling step is selected based on a cost function.

Preprocessing

For computational efficiency the sample domain is constrained to those voxels with a feasible gradient magnitude, after regularizing with a simple Gaussian filter. An unseen 3D MR volume I is reduced to a constrained sample domain by the following binarization process

$$I_{\nabla} := \begin{cases} 1, & \text{if } |\nabla\{I * g\}| \geq \mu_{\{|\nabla\{I * \boldsymbol{g}\}|>0\}} \\ 0, & \text{else,} \end{cases} \tag{3.23}$$

where ∇ is the gradient operator, $\boldsymbol{g}$ denotes a Gaussian kernel, $*$ is the convolution operation, and $\mu_{|\nabla\{I*\boldsymbol{g}\}|>0}$ represents the mean of all positive values in the smoothed gradient magnitude volume $|\nabla\{I * \boldsymbol{g}\}|$. Fig. 3.10 (a) shows an exemplary axial MRI slice, in which the spherical femoral head is visible. Fig. 3.10 (b) shows the corresponding axial slice from I_{∇}. Since anatomical structures that contain spherical components are the objects of interest, it is possible to further restrict the sampling domain, by only keeping those image points, that are most likely part of the femur head.

As the intersection of a sphere with hyper planes of arbitrary orientation always results in a circle on that plane, a canonical strategy is the application of 2D Hough transforms on axial, frontal, and sagital volume slices. For each slice in each direction the $n_{circles} > 0$ most probable circles are considered, respectively. The intersection of circles with different spatial orientation yields voxel coordinates that presumably address spherical structures. The more orientations are involved in the intersection, the higher is the probability of the intersection being part of an actual spherical structure. Fig. 3.10 (c) shows a heat map of probable sphere points, where the intensity reflects the number of circles of different orientation involved in this intersection point. The number of intersections from circles of different spacial orientation is stored in a voting volume, rendering the heat maps. For the 2D circle detections the minimal and maximal radii, and the standard deviation $r_{min}, r_{max}, \sigma_r$ from the GESAMs model $\mathcal{G}$ are used to limit the radius range of the 2D Hough transforms to $[r_{min} - \frac{\sigma_r}{2}, r_{max} + \frac{\sigma_r}{2}]$.

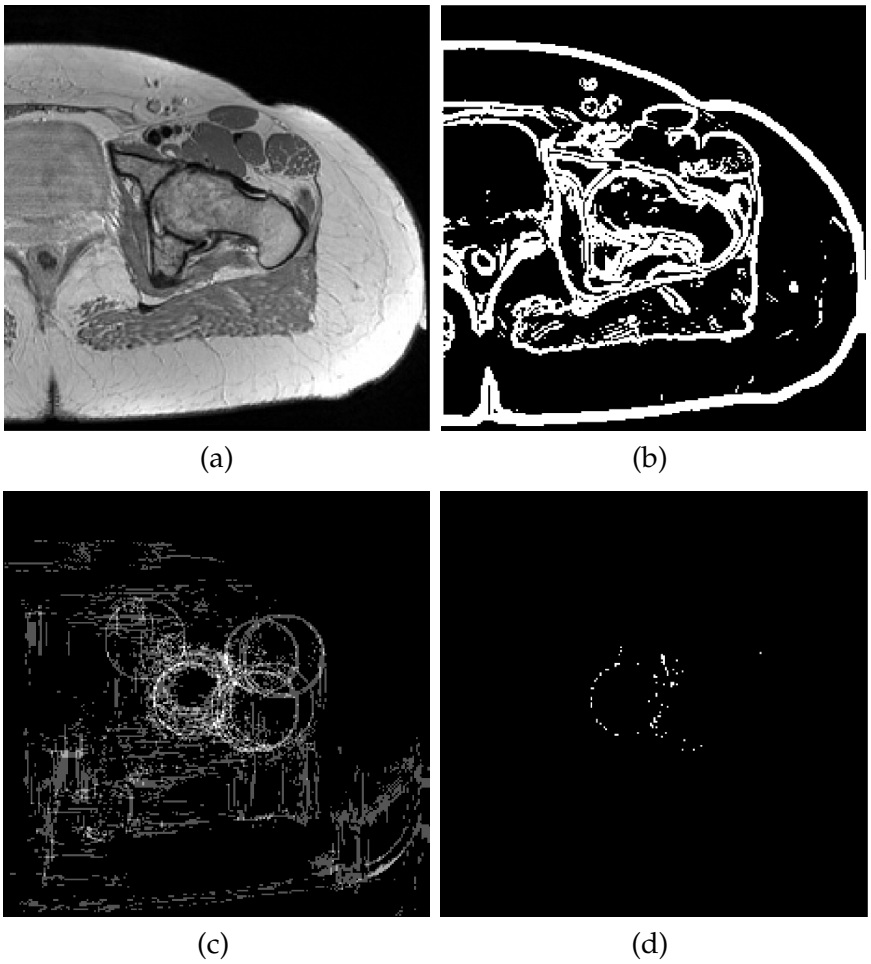


FIGURE 3.10: Illustration of preprocessing steps. (a) Axial slice of MRI data. (b) Corresponding binary slice of voxels with strong gradient, taken from I_∇. (c) Respective slice of voting volume of intersecting circles with different spatial orientations. (d) Axial slice of $I_{\nabla,\circ}$.

Let $I_\circ$ denote the binary volume, containing the most probable aforementioned intersections. Then the Hadamard product

$$I_{\nabla,\circ} := I_\nabla \cdot I_\circ$$

contains volume points that both have a strong gradient and also probably contribute to a feasible sphere hypothesis. Any volume point p in $I_{\nabla,\circ}$ that has a value of $I_{\nabla,\circ}(p) = 1$ will be referred to as a *feasible sample point*. Fig. 3.10 (d) depicts the feasible sample points of the corresponding axial slice in $I_{\nabla,\circ}$, achieved by the preprocessing procedure described in this section.

Structured Sampling

With the drastically reduced sample set in $I_{\nabla,\circ}$, a structured sampling approach is proposed, to ensure the suggestion of promising sphere candidates. For each k_{sample}-th feasible sample point p, with $k_{sample} \geq 1$, a 3D sampling cube with width $r_{max} + \frac{\sigma_r}{2}$ is spanned around p. From this restricted sampling cube three additional feasible sample points are randomly chosen to propose a sphere candidate. This process is repeated until at least a minimal number of sufficient sphere proposals $n_{sp} \geq 1$ with

a radius within the range $[r_{min} - \frac{\sigma_r}{2}, r_{max} + \frac{\sigma_r}{2}]$ is proposed. If this is not possible, a maximum number of subsequent non-sufficient proposals $n_{nsp} \geq 1$ ensures termination.
In Fig. 3.11 the proposed sampling strategy is visualized, where the current sample point p is depicted in blue. The yellow points represent the additional feasible sample points, that are randomly selected from all sampling points (red) within the 3D sampling cube. The selected points and the current sample point p uniquely define a sphere candidate.

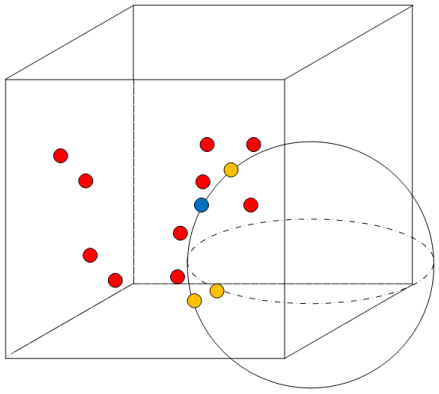


FIGURE 3.11: Illustration of structured sampling strategy. A cube is spanned around the current sampling point (blue). Three additional feasible sample points (yellow) are randomly selected to define a distinct sphere. Remaining feasible sample points are depicted red.

Best Sphere Selection

Based on the sphere candidates from the previous step, the most probable candidate needs to be selected for the final localization of the near-spherical component, i.e. the femur head, within the unseen MR volume I. For simplicity, only the center point p_{sc} will be used to represent a sphere candidate in the following sections, although a sphere candidate is actually characterized by both its center point position and its radius. If the sphere candidate's radius is of relevance it will be referred to as $r(p_{sc})$. The sphere evaluation is based on a combination of various selection criteria. In the context of femur head localization in MR volumes, the following criteria are considered:

- **Similarity of inner intensity distribution**
  Similar to Eqs. (3.14,3.17) the intensity distribution within the sphere candidate can be represented by a feature vector $v(p_{sc})^{in}$, where p_{sc} denotes the sphere candidate's center point. With $\mathscr{M}^{in}$ from the training stage, similarity to the training atlases can be measured by the squared Mahalanobis distance [DMJRM00]

$$s^{in}(p_{sc}) := \sum_{i=0}^{n_{PCA}-1} \frac{1}{\lambda_i^{in}} \left({u_i^{in}}^T \cdot (v(p_{sc})^{in} - \overline{v}^{in}) \right)^2 .$$

- **Similarity of outer intensity distribution**
  Following the same intuition, the similarity between the intensity distribution

of the outer boundary $v(p_{sc})^{out}$ to the distributions from the training data, encapsulated in $\mathscr{M}^{out}$ is measured by

$$s^{out}(p_{sc}) := \sum_{i=0}^{n_{PCA}-1} \frac{1}{\lambda_i^{out}} \left(u_i^{out^T} \cdot (v(p_{sc})^{out} - \overline{v}^{out}) \right)^2 .$$

- **Similarity of inner and outer intensity distribution**
  The concatenation of inner and outer intensity feature vectors for a sphere candidate with center point p_{sc} yields a joint feature vectore $v(p_{sc})^{joint}$. Its similarity to the observations in the training data can be again measured by means of the squared anobis distance, using $\mathscr{M}^{joint}$, i.e.

$$s^{joint}(p_{sc}) := \sum_{i=0}^{n_{PCA}-1} \frac{1}{\lambda_i^{joint}} \left(u_i^{joint^T} \cdot (v(p_{sc})^{joint} - \overline{v}^{joint}) \right)^2 .$$

- **Scaled Number of Inliers**
  Sample points that are within a small error margin, and can be modeled with the fitted model, are called *inliers*. In case of sphere fitting, the inliers consist of sample points that lie within a narrow band, surrounding the sphere candidate's boundary. In the conventional RANSAC procedure, the number of these inliers is used as an indicator of how suitable the current model is. Fig. 3.12 illustrates different example sphere candidates with their corresponding inliers within the narrow band.
  Let $n_{inlier}(p_{sc})$ denote the number of sample points within the narrow band of width $\varepsilon > 0$ of a sphere candidate with center point p_{sc}. Since large sphere candidates tend to have a larger absolute number of inliers, because the narrow band has a larger volume, the number of inliers needs to be scaled using the sphere candidate's radius $r(p_{sc})$, i.e.

$$n_{inlier,scaled}(p_{sc}) := \frac{n_{inlier}(p_{sc})}{4\pi r(p_{sc})^2}.$$

  Since $\varepsilon > 0$ is assumed to be very small, the number of inliers correlates with the sphere surface area. Therefore, a scaling factor of $4\pi r(p_{sc})^2$ is utilized. With $n_{inlier,scaled}(p_{sc})$, sphere candidates of different volumes can be fairly compared against each other.

- **Center Point Distance to Mean of Inliers**
  Another considered criterion is the distance of the sphere candidate center point to the mean of all inlier sample points within the narrow band. Since the femur head is of near-spherical shape, the inlier sample points should mostly be equally distributed around the sphere candidates boundary. If the inliers are not equally distributed, their mean shows a deviation from the sphere candidate's center point p_{sc}. This is demonstrated in Figs. 3.12 (a)-(b). Let $\mu_{inliers}(p_{sc})$ denote the mean of all inlier sample points for a sphere candidate with center point p_{sc}. Then the distance of p_{sc} to $\mu_{inliers}(p_{sc})$ is calculated by means of the Euclidean distance, i.e.

$$d_{inlier}(p_{sc}) := ||p_{sc} - \mu_{inliers}(p_{sc})||_2.$$

- **Homogeneity within Sphere Candidate**
  For the particular case of femur head localization via sphere fitting, the homogeneity of the sphere candidate's encapsulated area can be used to assess, whether the proposed candidate is feasible or not. The homogeneity of the area is estimated using the mean gradient magnitude of the MR volume. Let p_{sc} denote the sphere candidate's center point and ∇I a volume representing the intensity gradients of the MR volume I. Moreover, let

  $$S^{in}_{p_{sc}} : \Omega \to \{0, 1\}$$

  be the indicator function, of whether an arbitrary point $p \in \Omega$ is encapsulated by the sphere candidate with center point p_{sc}. Then the homogeneity $\eta(p_{sc})$ of the sphere candidate with center point p_{sc} is estimated by

  $$\eta(p_{sc}) := -\frac{\sum_p |\nabla I(p)| \cdot S^{in}_{p_{sc}}(p)}{\sum_p S^{in}_{p_{sc}}(p)}.$$

  The higher the homogeneity, the more likely it should be that the sphere candidate is located at the femur head.

- **Intensity Variance within Sphere Candidate Boundary**
  Similar to the homogeneity of the inner sphere candidate region, the intensity variance of the outer boundary region may be used as an indicator of whether the sphere candidate is feasible. The outer boundary region is estimated to have the same volume as the inner sphere region, according to Eq. (3.10). Let p_{sc} again denote the sphere candidate's center point and I the MR volume. Furthermore, let

  $$S^{out}_{p_{sc}} : \Omega \to \{0, 1\}$$

  be the indicator function, of whether an arbitrary point $p \in \Omega$ belongs to the outer boundary region of the sphere candidate with center point p_{sc}. Then the intensity variance $\sigma^2(p_{sc})$ of the sphere candidate with center point p_{sc} is estimated by

  $$\sigma^2(p_{sc}) := \frac{\sum_p (I(p) - \mu_I(p_{sc}))^2 \cdot S^{out}_{p_{sc}}(p)}{\sum_p S^{out}_{p_{sc}}(p)},$$

  where

  $$\mu_I(p_{sc}) := \frac{\sum_p I(p) \cdot S^{out}_{p_{sc}}(p)}{\sum_p S^{out}_{p_{sc}}(p)}$$

  depicts the mean intensity of the outer boundary region of the sphere candidate with center point p_{sc} in I. Since the femur head is surrounded by various tissue types, e.g. bone, muscle, cartilage, it is assumed, that the intensity variance is rather high.

- **Number of Dead Edge Points**
  The last criterion for sphere candidate selection, considered in the scope of this work, is the number of *dead* edge points. An edge point that lies within the sphere candidate is considered a *dead* edge point. While it is encapsulated by the sphere candidate, it does not contribute in increasing the number of inliers.

The number of dead edge points for a sphere candidate with center point p_{sc} will be denoted as $n_{dead}(p_{sc})$. For a feasible sphere candidate, there should only be few dead edge pixels, as illustrated in Figs. 3.12(a) and (c). In fact, this can be considered a more specific version of the homogeneity aspect.

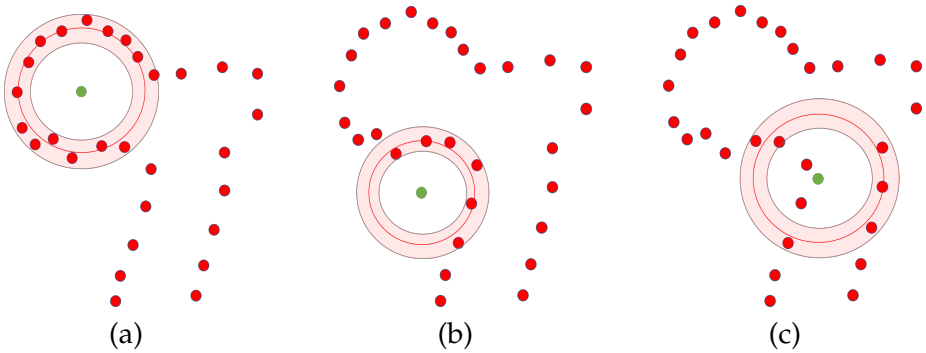


FIGURE 3.12: Hypothetical sphere candidates with narrow band and inliers. (a) The sphere candidate is correctly positioned at the femur head. (b) A sphere candidate example, in which the inliers are not equally distributed along the sphere boundary. (c) A sphere candidate with multiple dead edge sample points within the candidate's inner region.

Altogether, sphere candidates are evaluated based on a weighted combination of the aforementioned criteria. To achieve better comparability between the criteria, each criterion is scaled by its maximally achieved value by the proposed sphere candidates from the structured sampling step. Let P_{sc} denote the set of the center points of all proposed sphere candidates from the structured sampling step. For a sphere candidate with center point p_{sc}, its cost function is then defined as

$$
\begin{aligned}
c(p_{sc}) := \; & w_{s^{in}} \cdot \frac{s^{in}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} s^{in}(p'_{sc})} \\
& + w_{s^{out}} \cdot \frac{s^{out}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} s^{out}(p'_{sc})} \\
& + w_{s^{joint}} \cdot \frac{s^{joint}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} s^{joint}(p'_{sc})} \\
& - w_{n_{inlier}} \cdot \frac{n_{inlier,scaled}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} n_{inlier,scaled}(p'_{sc})} \\
& + w_{d_{inlier}} \cdot \frac{d_{inlier}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} d_{inlier}(p'_{sc})} \\
& + w_{\eta} \cdot \frac{\eta(p_{sc})}{\min\limits_{p'_{sc} \in P_{sc}} \eta(p'_{sc})} \\
& - w_{\sigma^2} \cdot \frac{\sigma^2(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} \sigma^2(p'_{sc})} \\
& + w_{n_{dead}} \cdot \frac{n_{dead}(p_{sc})}{\max\limits_{p'_{sc} \in P_{sc}} n_{dead}(p'_{sc})}.
\end{aligned} \tag{3.24}
$$

A more detailed hyper parameter analysis for the weight configuration of the proposed criteria and an investigation of their impact on a satisfactory sphere selection is presented in section 3.4.4.

3.4.3 Expansion

As of now, the suggested near-spherical component only localizes the femur head. Information about the femur shaft location in relation to its head is, however, still missing. The third stage deals with the expansion of the detected near-spherical component into the remaining femur to determine the shaft location and therefore the femur's spatial orientation. This is achieved by a simplified region based Level Set approach (see chapter 2.2), in which gradient information of the volume is additionally used to heavily restrict the expansion to stay within the femur.
Chan and Vese [CV01] propose a global energy approach, where the inner and outer regions of the contour are considered. Following the notation from chapter 2.2 and given the Heaviside function

$$\mathscr{H}(\Phi(p)) = \begin{cases} 1 & \text{, if } \Phi(p) > 0 \\ 0 & \text{, else} \end{cases}$$

Chan and Vese propose a region based energy function

$$\begin{aligned} E_{CV}(\Phi) := \int_{\Omega} & \mathscr{H}(-\Phi(p))|(I(p) - \mu_{in}|^2 \\ & + \mathscr{H}(\Phi(p))|I(p) - \mu_{out}|^2 \, \mathbf{d}p, \end{aligned}$$

in which deviations from the mean intensities μ_{in}, μ_{out} of inner and outer region are punished for each inner and outer point, respectively. This yields a gradient descent based update rule

$$\Delta\Phi^t = -\nabla_{\Phi^t} E_{CV}.$$

Nevertheless, the outer area can be large and often contains many different structures, potentially providing misleading information for the contour deformation. Therefore, a restriction of the outer region to a band around the current contour is suggested, which is loosely based on Jung and Jung's work [JJ08].
Instead of applying the Level Set approach on the original MR volume, a dilated version of I_{∇} from Eq. (3.23) is used. The idea is that strong gradients serve as boundaries of the expansion, which are additionally strengthened by an additional dilation operation. This simplification of the volume therefore helps in keeping the expansion within the actual femur bounds. The located sphere from the localization stage serves as initial contour of the Level Set approach. Reducing the initial contour's radius ensures initialization within the femur. To increase computational efficiency, I_{∇} can be resized to a smaller size.

3.4.4 Experiments

The following paragraph describes the experiments conducted with the proposed GESAMs approach for femoral model initialization. First, the data sets, used for the experiments are presented. The subsequent segment presents the achieved initialization results in a leave-one-out cross validation setting compared to the *Elastix*

multi-atlas 3D registration baseline, used in section 3.3.5 (Multi-Atlas II). The last subsection presents a retrospective hyper parameter analysis regarding the cost function (see Eq. (3.24)), described in section 3.4.4.

Data

For the experiments, the same data sets were used as described in section 3.3.5. All patient volumes were resized to have an isotropic voxel spacing of $(1 \times 1 \times 1)mm^3$ to meet the prerequisites of GESAMs.

Leave-One-Out Model Initialization

In a leave-one-out cross validation manner, each available patient volume P1, . . . ,P6 and P1PO, P2PO serves as a test volume once, whereas the remaining patient volumes and their corresponding label volumes are considered training volumes and are used to form the GESAMs in the training stage. Following the procedure of the PAMs experiments (see section 3.3.5), only MR volumes are used for training, which are not of the same patient as the test volume. For instance, P1 is not used during training for the test volume P1PO.
In the training stage, the number of intensity bins, determining the feature length, is set to $n_{bins} = 20$. For preprocessing in the localization stage the number of proposed circles per slice in each orientation by the Hough Transform is set to $n_{circles} = 10$. For the proposed structured sampling method the step size k_{sample} is determined, such that about 500 feasible sample points remain in $I_{\nabla,\circ}$. Moreover, $n_{sp} = 10$ and $n_{nsp} = 10$ have been empirically determined as sufficient termination parameters. In the expansion stage, the volume size is reduced to a factor of 0.5 in each dimension for the Level Set approach, and is initialized with the located sphere reduced to half of the detected radius from the localization stage. Table 3.7 shows the preliminary weight configuration that is used for the published results in [Pha+19c]. A more detailed retrospective analysis is presented in section 3.4.4. The DSC is used to

$w_{s^{in}}$	$w_{s^{out}}$	$w_{s^{joint}}$	$w_{n_{inlier}}$	$w_{d_{inlier}}$	w_η	w_{σ^2}	$w_{n_{dead}}$
2	10	5	0	1	0	0	0

TABLE 3.7: Weight configuration for experiments.

estimate and compare the localization quality. In the baseline multi-atlas approach, majority voting is conducted on the weighted summation of the transformed label images. Here, the optimization metric for the registration is used as weight. For each data set, the binarization threshold $\zeta \in \{0.1, 0.2, \ldots, 1.0\}$ resulting in the best DSC for the multi-atlas segmentation, is chosen to compare to the GESAMs results. Table 3.8 shows a detailed overview of the achieved DSCs for each data set, comparing the multi-atlas localization quality with the achieved results of the proposed GESAMs approach before and after the expansion stage. An average DSC value of 0.5416 ± 0.1450 is achieved by the multi-atlas approach, which is surpassed by the GESAMs with a mean DSC value of 0.7507 ± 0.0108 after expansion. It is also noticeable, that the GESAMs approach outperforms the registration method for every test patient.
As discussed in section 3.3.5, the weaker multi-atlas performance may be due to the different field of views (FOVs) of the MR volumes. The inconsistency of the FOVs increases the difficulty of the registration problem. Although the GESAMs approach requires the femur to have a near spherical component, i.e. its femur head,

DSC	Multi-Atlas II	GESAMs before Exp.	GESAMs after Exp.
P1	0.6919	0.4205	0.7334
P1PO	0.4974	0.4583	0.7476
P2	0.7373	0.4989	0.7687
P2PO	0.6586	0.4674	0.7510
P3	0.5840	0.4335	0.7634
P4	0.1562	0.4890	0.7626
P5	0.6161	0.4068	0.7282
P6	0.3912	0.3805	0.7510
$\emptyset$	0.5416 ± 0.1450	0.4444 ± 0.0340	0.7507 ± 0.0108

TABLE 3.8: Resulting DSCs from proposed GESAMs approach before and after the expansion stage compared to the multi-atlas approach. Revised from [Pha+19c].

it nevertheless poses a viable alternative to registration approaches, which are often sensitive to varying FOVs.

Eligibility of GESAMs

In an additional experiment, the necessity of the training phase to generate the GESAMs is investigated. Since the preprocessing phase in the localization stage may already reduce the sampling domain to feasible sample points, that mostly lie on the femur head's boundary region, it is possible to argue that this information may already be sufficient to approximate the femur head's location without the need of a prior training phase. It should be, however, noted that the preprocessing stage already requires information from the GESAMs, as the slice-by-slice 2D Hough transforms are constrained by the radii acquired from the training data.

In the additional experiment, artificial feasible sample points, which form a perfect sphere outside the femur head region, are induced into the drastically reduced sampling domain $I_{\nabla,\circ}$. This simulates an anatomical structure with near spherical shape, that is different from the femur head. The same weight configuration as in Table 3.7, which will be denoted as *GESAMs config*, is compared to a weight configuration, in which only the number of inliers n_{inlier} is considered, thus neglecting any GESAMs information for the best sphere selection step in the localization stage. This configuration will be referred to as *Naive config*. In the same leave-one-out cross validation manner as described in section 3.4.4, each available patient volume is considered as a test volume once. Since the proposed structured sampling step yields a randomized component, the experiment is performed 100 times for each test patient. For this experiment the *feasible sphere proposal rate (FSPR)* is calculated for both weight configurations. The FSPR is the percentage of feasible sphere proposals, where a sphere proposal is considered *feasible*, if the proposed sphere center lies within the actual femur head.

Table 3.9 depicts detailed overview of the localization performances for each data set in both configurations. It becomes apparent, that for the data sets $P3$ and $P6$ the naive configuration achieves better localization rates than for the other data sets. This might however be due to the more spherical nature of these data set's femur head. For the naive inlier based configuration, about 22.13% of the time a feasible sphere is proposed, whereas the GESAMs based configuration results in a FSPR of

FSPR in %	P1	P1PO	P2	P2PO	P3	P4	P5	P6	∅
Naive config	0	0	0	7	68	5	31	66	22.13 ± 24.66
GESAMs config	94	84	96	94	90	100	100	100	94.75 ± 4.25

TABLE 3.9: Resulting FSPR for a naive configuration without any GESAMs information compared to GESAMs configuration according to Table 3.7.

about 94.75%.

Fig. 3.13 depicts the additional artificial feasible sample points, that form a perfect sphere outside the femur head region. The resulting sphere localization before expansion by means of the GESAMs based configuration is visualized in yellow. It becomes apparent, that the femur head is correctly detected in the sphere selection stage, although a more spherical structure is present in the scene.
All in all, these observations indicate that the GESAMs based sphere proposal is able to robustly locate the femur head by leveraging the learned appearances from the training phase, even if other more spherical structures are present in the MR volume, whereas naive sphere detection algorithms canonically tend to locate the most spherical structure.

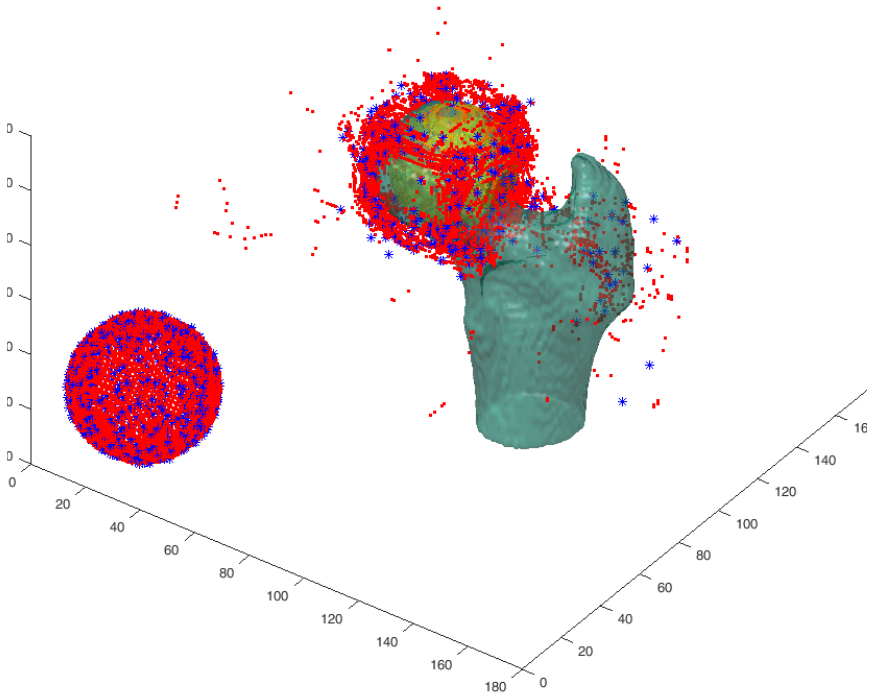


FIGURE 3.13: Overlay of feasible sample points (red dots), every k_{sample}-th sample point in the structured sampling stage (blue stars), GESAMs result before expansion (yellow), and ground truth (green). Artificially added feasible sample points form a perfect sphere.

Retrospective Hyper Parameter Analysis

In this subsection the weight configuration for the cost function in the sphere selection process (Eq. (3.24)) is investigated. Like in section 3.3.5, a grid search for the best weight configuration is applied, where the achieved DSC before the expansion step is used as quantitative metric. Again, there are 8 components to be assessed, so the weights are taken from a grid of natural numbers ranging from $\{0, \ldots, 8\}$, that sum up to eight. Fig. 3.14 shows a color encoded parallel plot, representing the results of the grid search. On the x-axis the proposed cost function components are depicted. The considered weights are depicted on the y-axis. The last two pillars on the right denote the achieved mean DSC over all test patients. Like in the previous section, each path from left to right represents one of the considered weight combinations, where a color close to red indicates a good overall performance. A color close to blue represents overall weak initialization results. It becomes imminent, that for a successful sphere detection stage, which is crucial for the subsequent stages, the cost components regarding the similarity of inner, outer, and joint intensity features are substantial, which is demonstrated in the parallel plot by the rather red incoming and outgoing edges at the top of the corresponding pillars. The variance within the sphere boundary on the other hand seems to be neglectable.

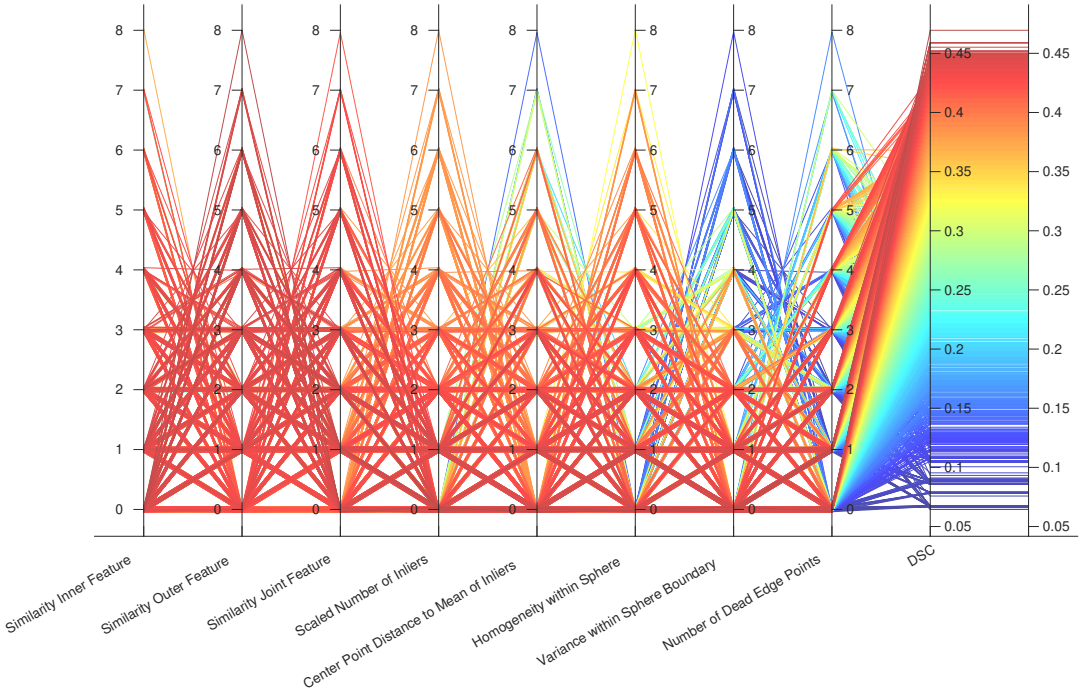


FIGURE 3.14: GESAMs parallel plot.

Tab. 3.10 allows a closer look at the best weight configuration for each patient. It is directly observable that for each patient a strong weight on s^{in} results in the most promising femur localization. The mean of the best weight configurations along all patients (second to last row) illustrates the importance of this cost component with a mean weight of 5.125 for s^{in}. Interestingly, the best mean DSC (last row) is accomplished by heavily weighting s^{out}, i.e. the similarity of outer intensity distributions. This is very similar to the originally considered configuration (see Tab. 3.7). The second best mean DSC of 0.4677 is however achieved by heavily weighting both similarities of inner and outer intensity distributions with weights of $w_{s^{in}} = 3$ and

$w_{s^{out}} = 4$.
Altogether, the more detailed inspection of the best configurations for each patient supports the observations made in the parallel plot, i.e. a strong focus on the learned appearance model based features s^{in} and s^{out} is of crucial importance.

	DSC	$w_{s^{in}}$	$w_{s^{out}}$	$w_{s^{joint}}$	$w_{n_{inlier}}$	$w_{d_{inlier}}$	w_{η}	w_{σ^2}	$w_{n_{dead}}$
P1	0.5008	8	0	0	0	0	0	0	0
P1PO	0.5092	6	0	1	1	0	0	0	0
P2	0.4711	7	0	0	0	0	0	1	0
P2PO	0.4696	6	0	0	0	0	0	0	2
P3	0.4773	2	4	2	0	0	0	0	0
P4	0.5202	3	1	2	0	1	1	0	0
P5	0.4230	6	0	0	0	0	0	0	2
P6	0.4540	3	0	1	1	0	0	0	3
$\emptyset$	0.4782	5.125	0.625	0.75	0.25	0.125	0.125	0.125	0.875
Best Overall	0.4680	1	6	0	0	0	0	0	1

TABLE 3.10: Best weight configuration for each patient. $\emptyset$ denotes the mean over all patients. *Best Overall* denotes the specific configuration, resulting in the best *mean* DSC.

3.5 Integration of Initialization Methods into Segmentation Pipelines

In this section, the previously presented femur initialization methods are integrated into a fully automated segmentation pipeline. Although multi-atlas registration methods can be considered fully automated end-to-end segmentation methods, their initial registration results are used as a baseline initialization for the subsequent segmentation approach. In the scope of this thesis, a Level Set approach is used as traditional segmentation method. Particularly, Paragios et al.'s variant of gradient vector flow (GVF) based Level Sets [PMGR01] is employed.

	Multi-Atlas II		PAMs	
	inital	after LS	inital	after LS
P1	0.6919	0.7821	0.8646	0.8384
P1PO	0.4974	0.6126	0.8170	0.8176
P2	0.7373	0.7474	0.7031	0.8216
P2PO	0.6586	0.6604	0.6602	0.7969
P3	0.5840	0.5804	0.8242	0.8101
P4	0.1562	0.1295	0.7999	0.7805
P5	0.6161	0.6990	0.8081	0.8278
P6	0.3912	0.4415	0.9132	0.8463
$\emptyset$	0.5416 $\pm$ 0.1450	0.5816 $\pm$ 0.1484	0.7988 $\pm$ 0.0586	0.8174 $\pm$ 0.0162

TABLE 3.11: Segmentation pipeline results using PAMs.

Tab. 3.11 shows the DSC improvements from the initial contour to the final segmentation for both the Multi-Atlas based initialization and the PAMs based initialization. Here, *LS* is an abbreviation for the Level Set segmentation method, which

should not be confused with the simplified Level Set method in the expansion step of the GESAMs approach. One can observe, that for both initialization methods the overall DSCs improve after applying the Level Set segmentation method. The final segmentation based on the atlas-based initialization, however, does not reach the initial mean DSC of the PAMs.

Tab. 3.12, on the other hand, shows the DSC improvements from the initial contour to the final segmentation for the multi-atlas based initialization compared to the GESAMs based initialization. Since the GESAMs use an isotropic voxel spacing of $(1 \times 1 \times 1)mm^3$, the atlases are resized accordingly before registration. Again, for both initialization methods, the overall DSCs improve after applying the Level Set method. However in this case, the GESAMs' performance improves by a lot from 0.7507 to 0.8642, compared to the PAMs initialization's increase from 0.7988 to 0.8174.

	Multi-Atlas II isotropic		GESAMs	
	inital	after LS	inital	after LS
P1	0.6697	0.8136	0.7334	0.8629
P1PO	0.4757	0.5390	0.7476	0.8613
P2	0.7706	0.7584	0.7687	0.8887
P2PO	0.6909	0.6776	0.7510	0.8795
P3	0.4624	0.3890	0.7634	0.8543
P4	0.6466	0.7158	0.7626	0.8495
P5	0.6654	0.7670	0.7282	0.8554
P6	0.7207	0.6994	0.7510	0.8616
$\emptyset$	0.6378 $\pm$ 0.0844	0.6700 $\pm$ 0.1030	0.7507 $\pm$ 0.0108	0.8642 $\pm$ 0.0010

TABLE 3.12: Segmentation pipeline results using GESAMs.

In direct comparison, the GESAMs start with an inferior initialization according to the DSC, but yield significantly better final results than the PAMs. A possible explanation lies in the heavily restricted expansion stage, in which gradient information is used to ensure that the expansion stays within the femur. This seems to be a more favorable initialization for the subsequent Level Set approach. However, both initialization strategies, yield superior initial and final mean DSCs compared the revised baseline multi-atlas registration approach.

3.6 Conclusion

In this chapter, two femur initialization strategies are presented, that make use of prior knowledge about the primitive shape structure of the femur.
On the one hand, the four stage PAMs based initialization considers 3D volumes as a sequence of 2D axial slices and leverages the near-circular shape of the femur in these slices to extract possible femur region candidates. Its intensity distribution model is incorporated into the femur's center line extraction, which is used together with the PAMs' border transition model to estimate the femur's boundary in polar space. Both stages are modeled by a path finding problem, which is addressed by means of dynamic programming. In the final fourth stage ICP is used to register an

existing shape model into the border point cloud.

On the other hand, the three stage GESAMs based initialization works directly on the 3D MR volume. Based on its intensity distribution appearance model of the femur head, first the femur head is located by means of a sphere selection process, in which a RANSAC like structured sampling approach is employed on the preprocessed MR volume. From the estimated femur head, a heavily restricted and region based Level Set method is utilized to expand into the remaining femur area, determining its orientation.

It is demonstrated that both proposed initialization approaches outperform the refined *Elastix* baseline multi-atlas registration method, especially in the context of strongly varying FOVs. Regarding PAMs, it is shown that a single labeled MR volume already yields reliable initializations in most cases.
For both methods, a retrospective hyper parameter analysis reveals that the most important components of the cost functions are in fact the components making use of the learned appearance models.

The feasibility of the initializations is validated on the example of a GVF based Level Set method, as a representative for contour based traditional segmentation approaches. The Level Set method shows favorable results, when the proposed initializations are used, compared the the multi-atlas registration initialization baseline. All in all, the presented initialization methods are viable alternatives to conventional registration approaches and contribute in complementing existing traditional femur segmentation approaches towards full automation.

Chapter 4

Shape Priors in Deep Learning Architectures

In the previous chapter, primitive shape priors, particularly circles and spheres, are utilized in contour initialization methods. These are a crucial component to completing fully automated segmentation pipelines for many traditional segmentation methods. Recently, deep learning approaches have emerged to state of the art methods, that deliver end-to-end segmentation solutions. The final segmentation prediction can therefore be obtained directly by feeding the neural network architecture with an input image or volume without the necessity for any previous localization of the structure of interest. Thus, full automation is already implied by the end-to-end architecture design of these segmentation networks. In this chapter, the incorporation of anatomical priors therefore aims at improving the segmentation performance of deep learning methods, instead of complementing them towards full automation like in chapter 3.
Conventional deep learning architectures, as described in chapter 2.3.2, usually depend on large data sets to cover different variants of sample appearances. Data augmentation by means of image rotation, flipping, translation, etc. are needed to avoid overfitting and to allow generalization.
Medical images, however, are often captured following a standardized protocol, especially in case of MR, CT and x-ray imaging. Therefore, medical images show a lot more similarities in appearance than in natural images, even if the field of view (FOV) may vary.

Especially bones and organs show relatively little variability in shape and topographical relationship to one another. The general idea is to build deep learning architectures, that leverage this additional prior information about consistent shape appearance across patients, and about possible topographical relationships. The goal is to improve the general segmentation performance, also considering one-shot and domain adaptation settings. The content of this chapter is based on the following previous publications for which content reuse was permitted:

> **[PDP21a]** Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "A Differentiable Convolutional Distance Transform Layer for Improved Image Segmentation". In: *Pattern Recognition: 42nd DAGM German Conference, DAGM GCPR 2020, Tübingen, Germany, September 28–October 1, 2020, Proceedings 42*. Springer. 2021, pp. 432–444

[Pha+19a] Duc Duy Pham et al. "Deep learning with anatomical priors: imitating enhanced autoencoders in latent space for improved pelvic bone segmentation in MRI". In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE. 2019, pp. 1166–1169
Copyright ©2011 IEEE

[Kav+21] A Emre Kavur et al. "CHAOS challenge-combined (CT-MR) healthy abdominal organ segmentation". In: *Medical Image Analysis* 69 (2021), p. 101950

[PDP21b] Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "Using Anatomical Priors for Deep 3D One-shot Segmentation.". In: *BIOIMAGING*. 2021, pp. 174–181

[PDP20] Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "Liver segmentation in ct with mri data: Zero-shot domain adaptation by contour extraction and shape priors". In: *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*. IEEE. 2020, pp. 1538–1542
Copyright ©2011 IEEE

In this chapter, the findings of these publications are partially revised and put into relation to each other. In the scope of using shape priors as anatomical priors, three strategies are presented to enforce shape information into the training process of deep learning architectures.

- A cascaded differential convolutional distance transform for the application in deep learning architectures
- Imitating encoder based architectures to mimic latent representations of ground truth segmentation maps
- Shape contour infusion and selected color augmentation to abstract from intensity based features

The following chapter is structured as follows:
Section 4.2 presents an approach to incorporate shape information into deep learning architectures by means of a convolutional distance transform, which is by design differentiable and therefore applicable in any generic deep learning setting.
Afterwards, section 4.3 introduces an *Imitating Encoder - Enhanced Decoder* architecture, that aims at incorporating shape information into the learning process, making use of the learned latent representation of ground truth segmentation maps.
This architecture is extended in section 4.4 to include Oktay et al.'s idea of shape regularization of the prediction [OF+18]. This redesign is used in the *Combined Healthy Abdominal Organs Segmentation (CHAOS)* challenge competition [Kav+21], yielding superior performance for a cross-modal segmentation setting. Furthermore, its applicability in the one-shot segmentation use case is investigated in additional experiments. Finally, ablation studies are conducted on the extended architecture. This is done in a general segmentation setting on the example of femur extraction from x-ray images, in which enough training samples are available and no domain shift is assumed.
In the next section 4.5, however, the feasibility of using shape information in a zero-shot domain adaptation setting is analyzed. Since the imitating encoder design is

not applicable to this specific task (for further elaboration see section 4.5), the impact of Oktay et al.'s anatomically constrained CNN (ACNN)[OF+18] is investigated and compared to the influence of additional contour information and color augmentations.
Related work regarding shape priors, few-shot segmentation, and zero-shot domain adaptation in medical image analysis is presented in the following first section 4.1.

4.1 Related Work

With the success of deep learning strategies on natural images, convolutional neural networks (CNNs) have also been introduced in the medical image processing domain and show state of the art performance. Especially Ronneberger et al.'s U-Net [RFB15], discussed in chapter 2.3.3, proposed specifically for semantic segmentation tasks in medical images, has achieved a lot of attention, as this architecture yields impressive segmentation results in many medical applications compared to traditional approaches. Therefore, its general architecture has been modified in numerous ways in order to improve the segmentation quality even more, e.g. by extending the applicability to 3D volumes [ÇA+16], modifying the loss function [KC+18], or incorporating residuals [MNA16].

4.1.1 Shape Priors

As many anatomical structures usually show only small shape variations, recent research naturally focuses on incorporating shape priors into the segmentation process. Shape priors have already been used in traditional segmentation methods before. Rousson et al. [RP02] leverage the idea of shape representation by means of signed distance transforms, proposed by Paragios et al. [PRR02], to incorporate shape priors into the level set framework. Cremers et al. [CSS03] also base their work on the distance transform's shape representation to enforce shape priors. Naturally, incorporating the distance transform into deep neural networks is a plausible step to model inter-pixel relationships, as also noted in Ma et al.'s work [Ma+20]. Dangi et al. [DLY19] apply distance map regression in a multi-task learning setting for cardiac MR image segmentation. They propose a regularization framework by formulating an Euclidean distance map regression objective, that is pursued by a sub-network of their segmentation architecture. Bui et al. [Bui+19] propose a similar multi-task approach, in which the geodesic distance is approximated as a learning task for neonatal brain segmentation. Similarly, Navarro et al. [Nav+19] also include the learning task of distance transform approximation in their multi-task segmentation approach. In these contributions, however, the distance transform needs to be learned, since the implementation of the distance transform is often not differentiable. In this thesis, Karam et al.'s [KSH19] derivation of a convolutional distance transform approximation is incorporated into the deep learning context. A thorough literature research has not led to any similar attempt, in which an adhoc differentiable convolutional distance transform layer is proposed for deep segmentation networks.

A different approach to incorporate shape priors is usage of deep encoder-decoder architectures to refine the network prediction. Ravishankar et al. [RV+17] extend the U-Net by means of a pretrained shape regularization autoencoder network, that is applied on the output of the U-Net to correct its prediction to a segmentation with feasible shape. Oktay et al. [OF+18] propose a similar supervised approach, in which a pretrained autoencoder is used to incorporate shape priors into the deep

learning architecture. However, instead of merely correcting the initial segmentation output, they make use of the autoencoder's encoding component to regularize the weight adaptation process of a generic segmentation network during training, which is motivated by Girdhar et al.'s architecture for generating 3D representations of objects from 2D images [GF+16].
In the scope of this thesis, an architecture similar to Girdhar et al.'s TL network is proposed, which makes use of autoencoders' compression capabilities.

4.1.2 One-Shot Segmentation

A limiting factor for most deep learning strategies is the amount of data needed to sufficiently train deep learning models. Especially in the medical domain, labeled training data is scarce and expensive to acquire. As a result, one-shot and few-shot learning approaches have been developed for classification tasks in natural image settings ([KZS15; SB+16; SSZ17; VB+16]). There is, however, little research towards one-shot learning in segmentation tasks ([DX18; MBE18]), particularly for medical images. Therefore, a short excursion into the applicability of shape priors in the context of one-shot segmentation is made.

4.1.3 Zero-Shot Domain Adaptation

Regarding deep domain adaptation for medical semantic segmentation, a suitable strategy would be to first pretrain a convolutional neural network (CNN) in the source domain, and to fine-tune the network in the target domain with fewer examples in a supervised manner, afterwards. Ghafoorian et al. [Gha+17] demonstrate the efficacy of such procedure on the example of brain lesion segmentation. They show that employing transfer learning on deeper layers improves coping with the domain shift between different MRI protocols, such as T1- and T2-weighted MRI. In this strategy, annotated training samples are needed from both source and target domain.

Another approach is the generation of synthetic but realistic training data for the target domain. For this strategy, the source domain images are transformed into the target domain. The synthetic target domain images are then paired with the corresponding ground truth data from the source domain for supervised training. Jiang et al. [Jia+18] use tumor aware generative adversarial networks (GANs) to generate synthetic MR images from CT images. For supervised training of their segmentation network, they mix the synthetic MRI data with a small set of real MR images.

In the context of image synthesis, Zhu et al. [Zhu+17] introduce cycleGANs, that are capable of conducting image-to-image style transfer for unpaired source and target images. This is of particular interest in the medical setting, as MR images may be converted to pseudo CT images without the necessity of previous co-registration. Kamnitsas et al. [Kam+17] apply a slightly different procedure by designing a domain discriminator which steers their segmentation network into learning domain-invariant features by adversarial training. In their work they specifically focus on the domain shift between different MRI protocols. These approaches usually do not need to make use of any ground truth information of the target domain, although they require (unlabeled) samples of the target domain for the adversarial training.

There has, however, only been little research towards zero-shot domain adaptation (or domain generalization), in which no information about the target domain is available. Zhang et al. [Zha+20a] propose heavy data augmentation to achieve domain generalization to overcome the domain shift across data sets of the same modality and sequence. Similarly, Hesse et al. [Hes+20] apply style and intensity augmentation to bridge the domain shift between data sets of the same modality, but with different sequences. A thorough literature research has, however, not led to any domain adaptation attempt, in which the domain shift is addressed between different *modalities* without any prior target domain information. The proposed strategies in section 4.5 make use of prior knowledge about anatomical inter-patient similarities in between most imaging modalities, such as similar contour progressions and shapes. Motivated by Geirhos et al.'s findings [Gei+18], the segmentation network is generally steered away from texture based and therefore modality specific features to ensure generalization towards unseen modalities.

4.2 Shape Constraint with a Convolutional Distance Transform

Deep learning based supervised segmentation methods usually aim to minimize a loss term during training, which is often defined on pixel level, e.g. the dice loss (see Eq. (2.2) in chapter 2.4.1). This reduces the segmentation task to a pixel-wise classification task. In these kind of loss terms, the error of one pixel is not reflected in the error of another, as pixels are considered independent of each other.
In this section, the distance transform, a well-established form of shape representation, is presented as a possibility to incorporate shape information into deep learning architectures. Particularly, Karam et al.'s [KSH19] derivation of a convolutional distance transform approximation is used to construct a differentiable convolutional distance transform layer, that can be directly attached to any deep segmentation network. The content of this section has been previously published in [PDP21a] and has been revised for this section.

4.2.1 Methods

A distance map of a binary image yields the distance of each pixel to its closest foreground boundary pixel. If the considered pixel is a foreground pixel itself, its distance is therefore zero. Common applicable distances are the Manhatten and the Euclidean distance. Let I denote a binary image. The distance between two pixel positions $p_i, p_j \in \Omega$ in I is depicted as $d(p_i, p_j)$. The distance transform

$$D_I : \Omega \rightarrow \mathbb{R}_0^+ \tag{4.1}$$

for I can be defined in a pixel-wise manner as:

$$D_I(p_i) = \min_{p_j : I(p_j)=1} \{d(p_i - p_j)\} \tag{4.2}$$

A major advantage of this type of image representation is the provision of information about boundary, shape, and location of an object of interest. Comparing a binary segmentation mask to its corresponding distance transform, the latter contains distance information about the closest object boundary in every pixel, whereas a binary segmentation mask only holds information of whether the structure of interest is present or not. In Fig. 4.1 the differences in binary images and in Manhatten distance transforms are illustrated on a simple toy example, in which two pixel values are swapped. For the binary representation, one can notice that only the affected pixels yield a difference, whereas in the corresponding distance transforms, the simple swap has a larger impact on the distance transform's landscape.

Convolutional Distance Transform

The following derivation of the convolutional distance transform (CDT) can be found in Karam et al.'s work regarding fast distance transforms [KSH19]. For the derivation, only translation invariant distances are considered, i.e.

$$d(p_i, p_j) = d(p_i + p_k, p_j + p_k) \tag{4.3}$$

for any image positions $p_i, p_j \in \Omega$ and any translation $p_k \in \mathbf{R} \times \mathbf{R}$. To calculate the distance transform of a binary image I, for each pixel position the distance to its closest foreground pixel needs to be acquired. Thus, for a fixed pixel position

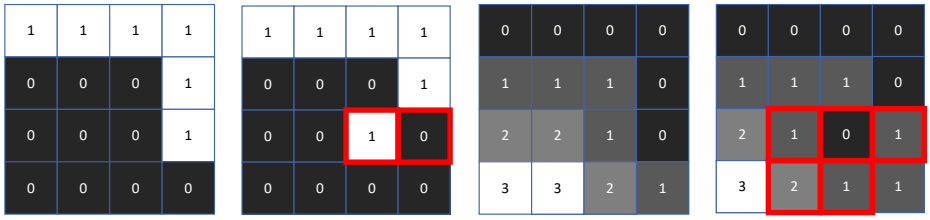


FIGURE 4.1: Differences in binary images (a) and (b) are only visible in the affected pixels, highlighted in (b). For the corresponding (Manhatten) distance transforms (c) and (d), differences propagate to further pixels, emphasized in (d), as these change the foreground shape and thus the distance landscape. Revised from [PDP21a].

the minimal distance to its closest foreground pixel is required. This can be accomplished by a minimum function over all possible distances from this position to any other foreground pixel. The minimum function can be approximated by a log-sum-exponential. Let $d_1, \ldots, d_n$ denote n distances, then the minimum function can be realized by

$$\min\{d_1, \ldots, d_n\} = \lim_{\lambda \to 0} -\lambda \ln \left(\sum_{i=1}^{n} \exp\left(-\frac{d_i}{\lambda} \right) \right). \tag{4.4}$$

The idea is that the exponential yields very small values the larger the distances are, as these are artificially increased by dividing by a very small $\lambda > 0$ and negated in the argument. Therefore, larger distances have a significantly smaller impact on the sum than small distances. In the extreme case, the exponential of large distances seeks zero, leaving only the exponential of the smallest distance in the sum. The subsequent natural logarithmic function $\ln(\cdot)$ then reverts the exponential operation, leaving an approximation of the minimum function, i.e.

$$-\lambda \ln \left(\sum_{i=1}^{n} \exp\left(-\frac{d_i}{\lambda} \right) \right) \approx -\lambda \ln \left(\exp\left(-\frac{d_1}{\lambda} \right) \right) = d_1 \tag{4.5}$$

for $1 > \lambda > 0$ and assuming that d_1 is the minimum. Here λ can be considered a parameter to determine the accuracy of the minimum approximation. The closer λ is to zero, the more accurate the minimum approximation becomes. With Eq. (4.4), it is possible to reformulate the distance transform of Eq. (4.2) to

$$D_I(p_i) = \lim_{\lambda \to 0} -\lambda \ln \left(\sum_{p_j : I(p_j) = 1} \exp\left(-\frac{d(p_i, p_j)}{\lambda} \right) \right). \tag{4.6}$$

Since a translation invariant distance is assumed, the distance between two points can be rewritten to

$$\begin{aligned} d(p_i, p_j) &= d(p_i - p_j, p_j - p_j) \\ &= d(p_i - p_j, 0) \end{aligned}$$

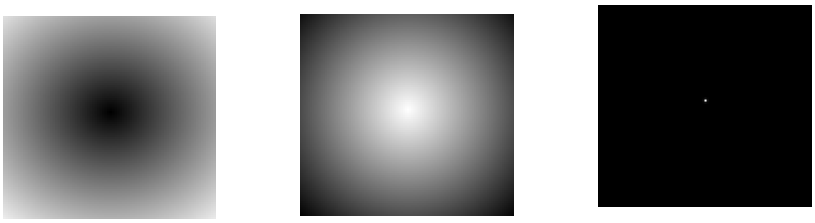
rendering a formulation of the distance transform as

$$D_I(p_i) \quad = \lim_{\lambda \to 0} -\lambda \ln \left(\sum_{p_j} I(p_j) \exp\left(-\frac{d(p_i - p_j, 0)}{\lambda} \right) \right), \tag{4.7}$$

which is the definition of a convolution. Thus, for a small $\lambda > 0$, the distance transform can be approximated by means of a convolution of the binary image I with a kernel $\exp\left(-\frac{d(\cdot,0)}{\lambda}\right)$, i.e.:

$$D_I \approx -\lambda \ln \left(I * \exp\left(-\frac{d(\cdot, 0)}{\lambda} \right) \right), \tag{4.8}$$

where $*$ is the convolutional operator.
Fig. 4.2 shows a visualization of $d(\cdot, 0)$, $-\frac{d(\cdot,0)}{\lambda}$ and $\exp\left(-\frac{d(\cdot,0)}{\lambda}\right)$ as discrete 2D kernels for a kernel size of 100. Since all operations in $\exp\left(-\frac{d(\cdot,0)}{\lambda}\right)$ are differentiable,



(a) $d(\cdot, 0)$ (b) $-\frac{d(\cdot,0)}{\lambda}$ (c) $\exp\left(-\frac{d(\cdot,0)}{\lambda}\right)$

FIGURE 4.2: Illustration of used kernel of size 100 for the convolutional distance transform.

this approximation can be directly integrated as a differentiable convolutional distance transform layer into current deep learning frameworks.

Cascaded Convolutional Distance Transform for Large Images

A major drawback of the convolutional design of the distance transform is that the kernel size theoretically needs to be twice as large as the the input image size. This ensures that even very sparse binary images can be transformed into a distance map by the proposed method. Otherwise background pixels that are not within the kernel reach of a foreground pixel would be assigned a distance of zero. This circumstance, however, yields the following two issues:

- The large kernel size leads to an increased computational complexity for the convolutional operation.
- For very large distances the exponential term for the kernel design in Eq. (4.8) may approach zero, decreasing the numeric stability of the logarithmic expression within the CDT. This issue particularly arises for large images with only few foreground pixels. Figure 4.3(c) shows the CDT of a toy example image (Fig. 4.3 (a)). It is clearly visible, that the CDT was only capable to calculate the distances for a specific range, before becoming unstable, in comparison with a standard Manhatten distance transform implementation in Fig. 4.3 (b).

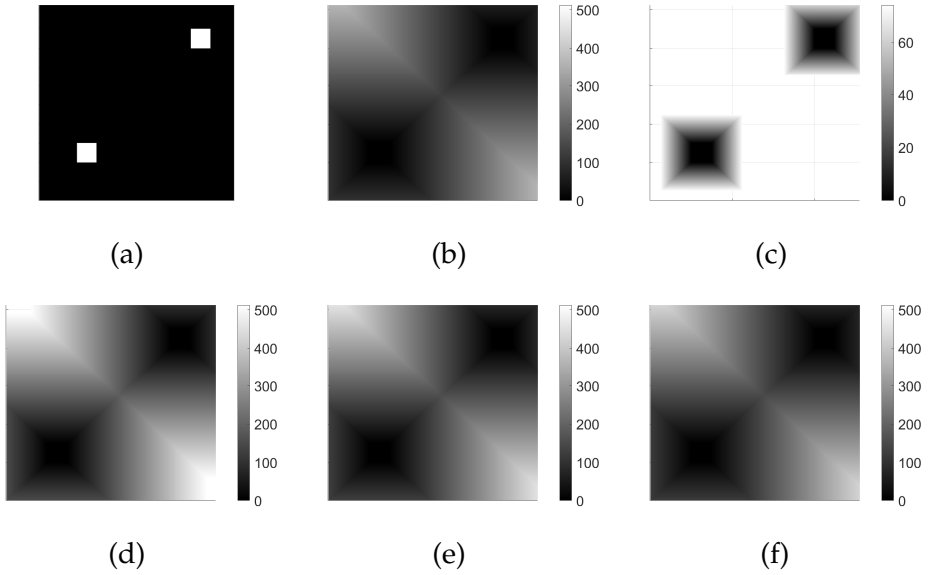


FIGURE 4.3: Limitations of the global CDT. (a) toy example of size 512×512, (b) standard Manhatten distance transform of (a), (c) global CDT of (a), (d)-(f) resulting cascaded CDTs of (a) with $k = 3, 5, 7$, respectively. Reused from [PDP21a].

A cascade of local distance transforms is presented to address the aforementioned issues of computational complexity and numerical instability. Instead of directly computing the distance transform with a large kernel, distance transforms with smaller kernels are cascaded to approximate the actual transform. Since the kernel size determines the maximal distance that can be measured, it is necessary to accumulate the calculated distances to form the final distance transform approximation.

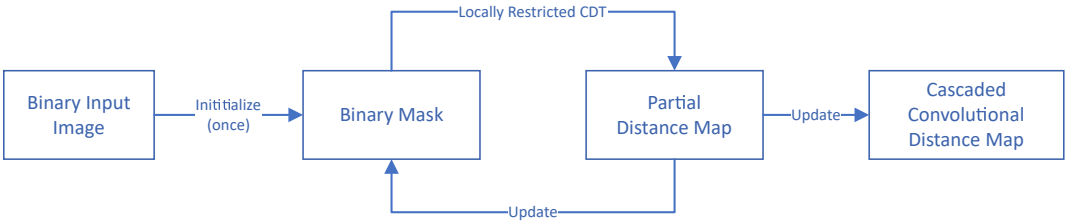


FIGURE 4.4: General concept of the cascaded convolutional distance transform.

Let k_{size} denote the kernel size. Then the maximal distance to a foreground point that can be captured by the CDT is limited to a range of $\lfloor \frac{k_{size}}{2} \rfloor$, considering the Manhatten distance. For all background points that are further away than $\lfloor \frac{k_{size}}{2} \rfloor$ from a foreground point, Eq. (4.8) yields a distance of 0, as within the kernel range there are only background points. The idea is to iteratively extend a binary mask by the area, for which a distance calculation was possible by the locally restricted CDT. This binary mask is initialized with the binary input image and serves the purpose of keeping track of which image parts have already been processed by the cascaded approach. The application of a locally restricted CDT on the binary mask results in a partial distance map, which is used to both update the binary mask by the newly covered area and to update the cascaded distance map result. Fig. 4.4 shows an

overview of the general procedure.
The regions from the partial distance map yielding a distance greater than zero, are used to update the binary mask by extending it by these non-zero regions. This extended binary mask can then be used to compute a new partial distance map. The calculated distances of the partial distance maps are combined with the distances of the previous iterations to form the final cascaded convolutional distance map.

For the i-th iteration, let $I^{(i)}$ denote the (updated) binary mask, and let $D_I^{(i)}$ denote the partial distance map, when applying the local CDT on $I^{(i)}$. For the i-th iteration, the original foreground area is assumed to have been widened by a margin of $i \cdot \lfloor \frac{k_{size}}{2} \rfloor$ in the updated binary mask from previous iterations. Therefore, this offset is additionally added to the currently estimated distances to compensate the lower kernel size. Thus, the cascaded distance map D_I^* is updated by the current distances by $i \cdot \lfloor \frac{k_{size}}{2} \rfloor + D_I^{(i)}$.

Without loss of generalization, let M denote the larger side of the input image I, i.e. $M \geq N$. Then at most $\lceil \frac{M}{\lfloor \frac{k_{size}}{2} \rfloor} \rceil$ of such local distance transforms are necessary to cover the whole image. Algorithm 1 summarizes this suggested procedure. Fig. 4.5

Algorithm 1 Cascaded Convolutional Distance Transform

function CASCADED_CDT(I, M, k_{size})
  $s \leftarrow \lceil \frac{M}{\lfloor \frac{k_{size}}{2} \rfloor} \rceil$ ▷ Calculate maximally necessary number of iterations
  $I^{(0)} \leftarrow I$
  $D_I^* \leftarrow I \cdot 0$ ▷ Initialize empty array for final cascaded distance map
  for i=0 **to** s **do**
    $D_I^{(i)} \leftarrow \text{CDT}(I^{(i)}, k_{size})$ ▷ Calculate partial distance map
    $I^{(i+1)} \leftarrow I^{(i)}$ ▷ Prepare update of binary mask
    for all $p : D_I^{(i)}(p) > 0$ **do**
      $D_I^*(p) \leftarrow D_I^*(p) + i \cdot \lfloor \frac{k_{size}}{2} \rfloor + D_I^{(i)}(p)$ ▷ Update cascaded distance map
      $I^{(i+1)}(p) \leftarrow 1$ ▷ Update binary mask
  return D_I^*

illustrates the proposed algorithm on an easy toy example and shows the updates of the binary masks, the corresponding partial distance maps, and the updated cascaded CDTs for the first three iterations $i = 0, 1, 2$.

The computational complexity of convolving an image of size $M \times N$ and a kernel with kernel size of k_{size} is $\mathcal{O}(M \cdot N \cdot k_{size}^2)$. For a global CDT the kernel size needs to be set to $k_{size} = 2 \cdot M$, rendering a complexity of $\mathcal{O}(M^3 \cdot N)$. The proposed procedure can drastically reduce the number of operations to $\mathcal{O}(M^2 \cdot N \cdot k_{size})$, if the kernel size is chosen much smaller than the image dimensions, i.e. $k_{size} << M$. Since the maximally possible measured distance of $d(\cdot, 0)$ in Eq. (4.8) is restricted by the kernel size, a small kernel size additionally yields a more stable computation of the logarithmic term as the exponential does not tend to approach zero.

Figures 4.3 (d)-(f) show the cascaded CDTs with kernel sizes of $3, 5, 7$, respectively. In comparison to the standard Manhatten distance transform in Fig. 4.3 (b), it becomes apparent that the offset assumption after each iteration yields an error that

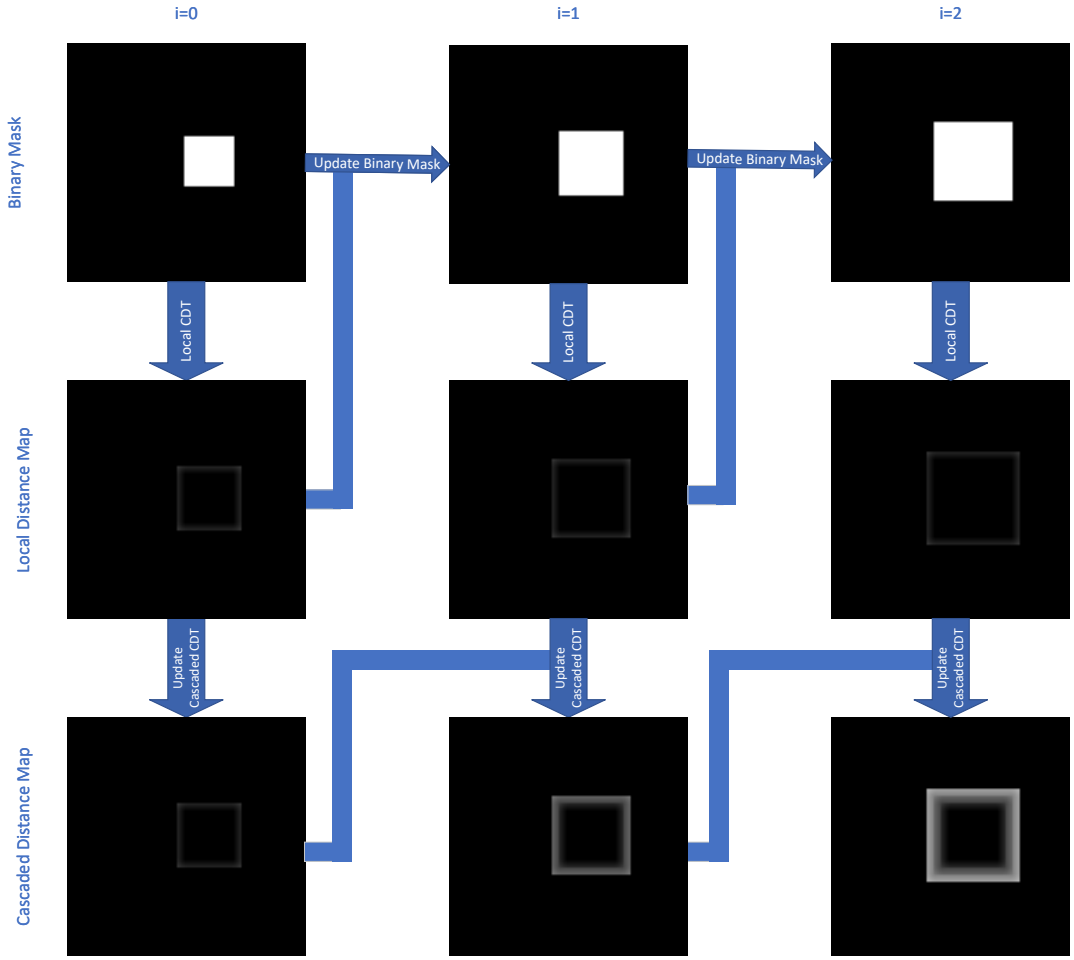


FIGURE 4.5: Illustration of the cascaded CDT Algorithm for the first three iterations $i = 0, 1, 2$ for a simple toy example.

is propagated to points with further distances. It is also visible, that this error decreases with increasing kernel size, as Fig. 4.3 (d) shows brighter areas, i.e. larger distances, than the reference in Fig. 4.3 (b), whereas Fig. 4.3 (f) shows less deviation from Fig. 4.3 (b). Thus, with the proposed procedure there is a trade-off that needs to be considered, namely between numerical stability by means of smaller kernel sizes and accuracy through larger kernel sizes. However, for the purpose of considering inter-pixel relationships in the weight optimization process of training CNNs, this approximation of the distance transform arguably suffices.

The Convolutional Distance Transform in Deep Learning

The previous subsection describes an adhoc cascaded convolutional approximation method of the distance transform for binary images. This approximation can be used to extend common segmentation networks, such as Ronneberger et al.'s U-Net [RFB15], in order to equip the segmentation loss with an additional regression loss, which compares the distance transform of the network's prediction with the distance transform of the ground truth.

Fig. 4.6 shows the general idea of how to extend the U-Net segmentation network with the proposed distance transform layer. In addition to the usual segmentation

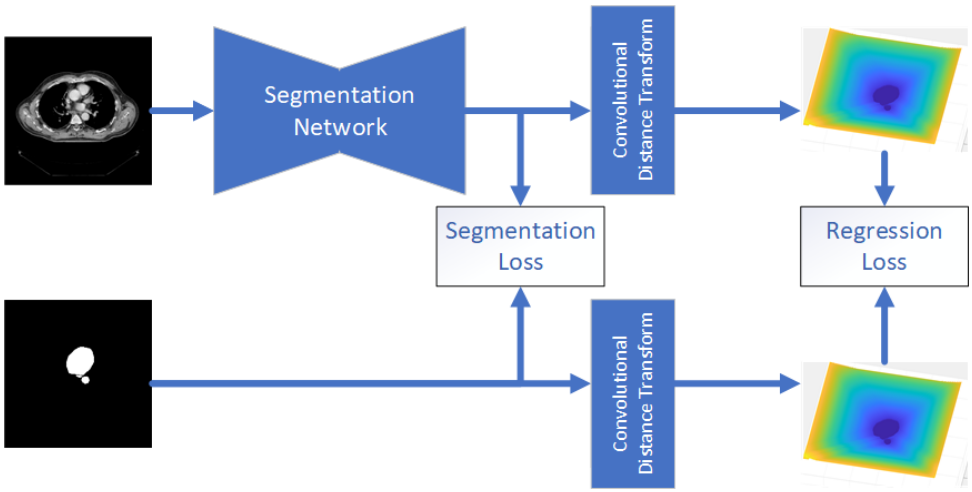


FIGURE 4.6: The convolutional distance transform layer can be attached to arbitrary segmentation networks. In addition to the segmentation loss, a regression loss of distance maps is calculated. Reused from [PDP21a].

loss, e.g. the dice loss, the predicted segmentation and the ground truth segmentation are both passed through the cascaded CDT layer to achieve the distance transforms of prediction and ground truth, respectively. These distance transforms contribute to a regression loss, e.g. the mean squared error, that considers inter-pixel relationships through the distance transforms. However, it needs to be noted that the segmentation's output is usually not binary. Assuming a final softmax or sigmoid activation, the output values for each channel vary between 0 and 1. For the restructuring of Eq. (4.6) to Eq. (4.7) it is however necessary to assume a binary image. A major challenge when blindly applying this approach to gray scale images lies in Eq. (4.7), as $I(p_i)$ may be a lot larger than the exponential for large distances, even if $I(p_i)$ is already very small. Therefore, even small probabilities of the segmentation output are considered in the sum and may be depicted as foreground pixels, distorting the actual distance map, as can be seen in Fig. 4.7. Here, the toy example from Fig. 4.3 is changed to a gray scale image (Fig. 4.7 (b)), in which the lower left square is set to a very low intensity of 0.001. The computed CDT of the binary image (Fig. 4.7 (c)) and the CDT of the grayscale image (Fig. 4.7 (d)) appear to be nearly identical, as can be also observed in their difference image Fig. 4.7 (f). The only differences of the computed distance transforms occur within the low intensity pixels, whereas the remaining distance transform's landscape does not show any changes. This is particularly problematic, as in the segmentation prediction even pixels with very low probabilities would then be considered as foreground pixels in the CDT.

This problem is addressed by proposing the following soft-threshold work around. Let C denote the number of classes, and let $y^{(c)}(p)$ be the prediction for class c at position p. Then a soft-threshold of the prediction can be achieved by

$$\tilde{y}^{(c)}(p) := ReLU\left(y^{(c)}(p) - \frac{C-1}{C}\right) \tag{4.9}$$

if C is small. This soft-threshold sets any prediction score below $\frac{C-1}{C}$ to zero. Therefore, strong and correct predictions are encouraged, as weak correct prediction scores are not registered for the distance transform and negatively impact the regression

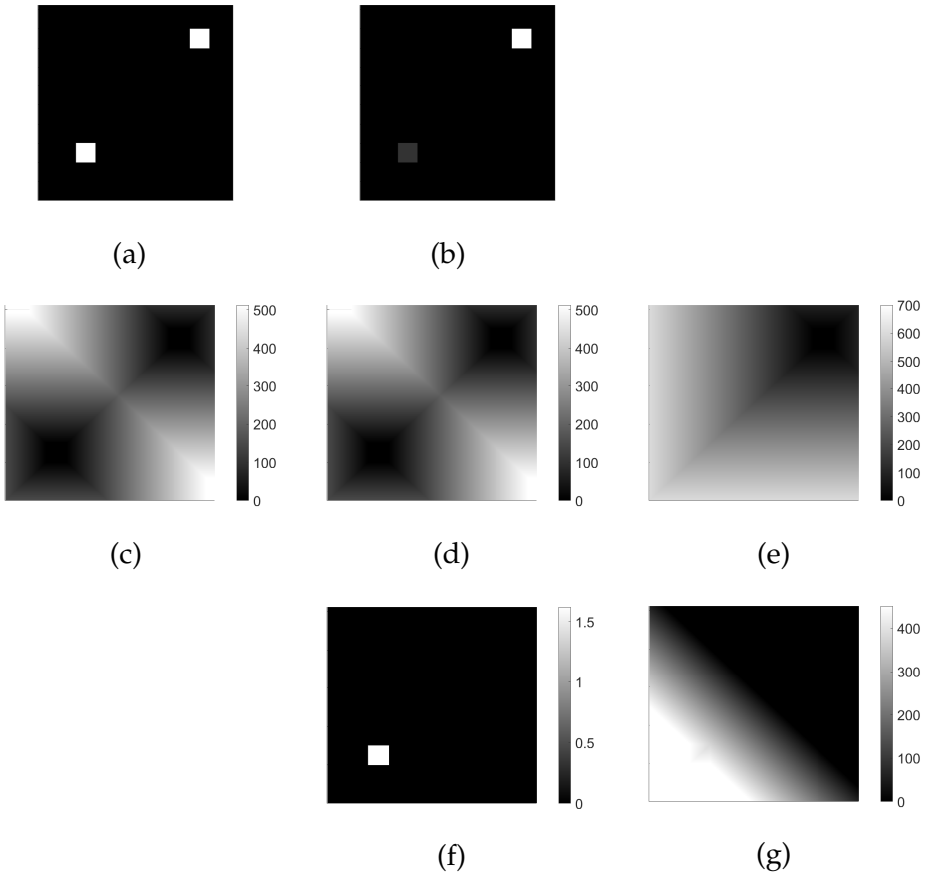


FIGURE 4.7: Limitations of CDT on gray scale images. (a) Binary mask, (b) Gray scale image, in which the intensity within the lower left gray square is set to 0.001, (c) Cascaded CDT of (a) as reference, (d) Cascaded CDT of (b), (e) Cascaded CDT of (b) with soft threshold, (f) Difference of (d) and (c), (g) Difference of (e) and (c). Reused from [PDP21a].

loss. Figure4.7 (c) shows the resulting CDT after applying the proposed soft-threshold, assuming two classes. Low intensity pixels are considered as background, so that a significant change in the distance transform landscape compared to the reference can be observed in Figure 4.7 (g).

4.2.2 Experiments

To investigate the impact of the proposed regression extension, ablation studies are conducted on the example of thoracic segmentation from CT scans of the thorax. For this a 2D slice-wise approach is utilized, in which the CT volumes are processed slice by slice by 2D networks.

Data

The publicly available SegTHOR CT data set [Lam+19] is used for the experiments. It consists of 40 CT volumes with corresponding ground truths of esophagus, heart, trachea, and aorta for training, as well as 20 CT volumes without available labels for testing. Since the goal of the ablation study is to investigate the influence of the presented cascaded CDT on the segmentation performance, outperforming optimized

ensemble methods within the SegTHOR challenge is refrained from. Instead, a valid comparison is achieved by training a U-Net architecture with and without the cascaded CDT on the same training and validation set to ensure fair comparability. In a hold-out validation manner, both models are trained on the 40 available training volumes, and the predictions are submitted for evaluation. Dice Similarity Coefficient (DSC) and symmetric Hausdorff Distance (sHD) are considered as evaluation metrics, which are both provided by the challenge's submission platform.

Implementation Details

A 2D U-Net and a variant including the cascaded CDT are implemented in Tensorflow 1.12. The CT volumes are fed slice-wise to the network and the slices are resized to an input size of 256×256. The U-Net implementation yields 5 scale levels with 2 convolutional layers, batch normalization and a max-pooling layer in each scale level. Starting with 32 kernels for each convolutional layer in the first size level of each contracting path, the number of kernels for each scale level is doubled on the contracting side and the number of kernels on the expansive side is halved for each change in scale level. A kernel size of 3×3 for every convolutional layer and 2×2 max-pooling is used in each architecture. For the U-Net optimization the standard dice loss $\mathscr{L}_{dice}$ (see Eq. (2.2) in chapter 2.4.1) is utilized as loss function. The mean squared error is used as regression loss $\mathscr{L}_{dist}$ for the cascaded CDTs of prediction and ground truth. The total loss function

$$\mathscr{L}_{total} := \mathscr{L}_{dice} + w_{dist}\mathscr{L}_{dist}$$

with weight $w_{dist} := 0.5$ to train the U-Net, which is equipped with the additional cascaded CDT. The optimization is performed with an Adam Optimizer with an initial learning rate of 0.001. The training slices are augmented by means of random translation, rotation and zooming. With a batch size of 4, both models are trained for 200 epochs and the model with best validation loss is chosen for evaluation. For the distance transform layer, $\lambda := 0.35$ is used, as suggested by Karam et al. [KSH19]. The experiments are conducted on a NVIDIA GTX 1080 TI GPU.

Results

Table 4.1 shows the achieved DSC scores of the trained models for esophagus, heart, trachea, and aorta. It is observable, that with the extension of the CDT the scores increase for all organs, except for the aorta. In this case the standard U-Net yields marginally better results. While the DSC score improves by more than 1% for the

DSC	Esophagus	Heart	Trachea	Aorta
U-Net	0.726312	0.809626	0.770571	**0.853953**
U-Net with CDT	**0.739028**	**0.822095**	**0.785847**	0.853011

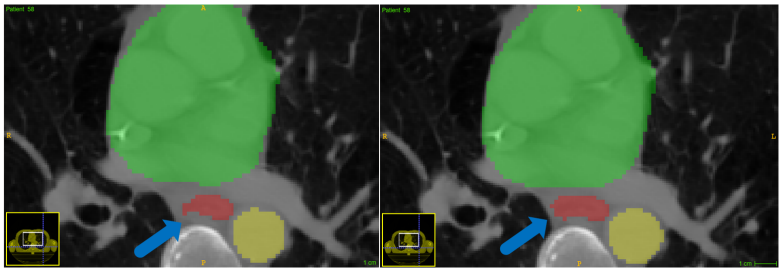
TABLE 4.1: Achieved DSCs for each organ. Reused from [PDP21a].

other organs, for the aorta the additional cascaded CDT regression does not seem to be beneficial. Wilcoxon signed rank tests [Wil92] with a significance level of 5% are applied on the achieved evaluation results to check, whether the improvements are of any significance. The improvements in DSCs for trachea and esophagus are significant with p-values of 0.0169 and 0.0040, respectively. For the DSC improvement in heart segmentation and the DSC difference in aorta segmentation, however, significance could not be established. The improvements can also be noticed in Table 4.2,

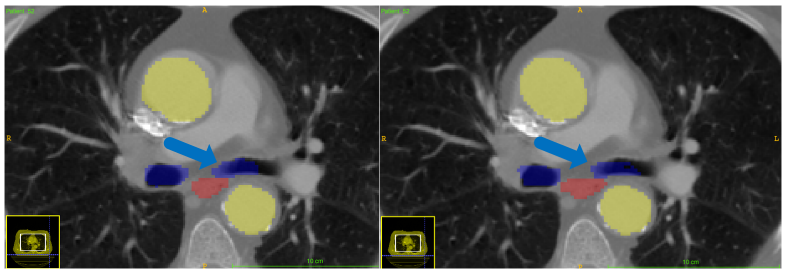
sHD	Esophagus	Heart	Trachea	Aorta
U-Net	1.511865	1.949374	2.137093	**1.900747**
U-Net with CDT	**1.113825**	**1.533211**	**1.649077**	2.004237

TABLE 4.2: Achieved symmetric Hausdorff Distances (sHD) for each organ. Reused from [PDP21a].

in which the symmetric Hausdorff distances are depicted. For esophagus, heart, and trachea the distances decrease with the proposed cascaded CDT regression, showing an improvement by approximately $25 - 30\%$. However, for the aorta a slightly worse mean distance is observed. This may be due to the fact, that the aorta seems to be a rather simple structure, that U-Net can already easily extract. The improvements in sHD are especially noteworthy, as a distance based regularization technique is used to improve the segmentation.
Regarding the Wilcoxon significance test, significant improvements for trachea, esophagus and heart with p-values of 0.0072, 0.0008 and 0.0400, respectively, can be observed. This underlines the assumption that the cascaded CDT regression adds significant value to more complex shapes, whereas simple structures as the almost circular aorta and heart slices are already well extracted by a standard U-Net. Fig. 4.8 shows exemplary segmentation predictions of both models on test data slices. The top images indicate better performance for esophagus segmentation with the proposed CDT, while the bottom images show superior segmentation results with the additional regression for the trachea. In both top and bottom row, the segmentation predictions of the aorta do not show noteworthy differences.



(a) U-Net (b) U-Net with CDT



(c) U-Net (d) U-Net with CDT

FIGURE 4.8: Exemplary segmentation predictions on the test set from both models, overlayed on considered image. The heart is depicted in green, the trachea in blue, the esophagus in red and the aorta in yellow. Top images indicate better performance for esophagus segmentation, bottom images for trachea. Reused from [PDP21a].

4.2.3 Summary

In this section, a novel cascaded convolutional distance transform is presented, which is differentiable and can therefore be used adhoc for any deep segmentation network without the necessity for prior training any distance transform component. The cascaded procedure reduces the computational complexity and overcomes numerical instability issues for large images. Additionally, a soft-threshold work around to address the demonstrated issue regarding non-binary segmentation outputs is presented. Ablation studies on the example of the segmentation of thoracic organs are conducted, in which the SegTHOR data set is used for training and evaluation. The experiments show promising results compared to an equally trained U-Net. Although marginal, the extension by the cascaded CDT yields significant improvements for most organs, particularly regarding sHD. These observations indicate, that a combination of proven non-deep learning concepts, such as the distance transform, and deep learning methods may yield great potential for future research.

4.3 Imitating Encoder - Enhanced Decoder Network

The previous section demonstrates the concept of incorporating shape prior information into the segmentation process by means of additional regularization that uses higher order information, represented by a distance transform. In the following section (and also its subsequent sections) convolutional autoencoders (CAEs) are employed as a possible alternative option to infer shape information by means of a learned latent representation of the ground truth segmentation. The general intuition is that ground truth information already implicitly contains anatomical information, which could be leveraged for the training process. The main idea is to compress that information by means of an autoencoder during training, to be able to incorporate this information for an increased segmentation performance. Instead of using pretrained networks to do so, a deep learning architecture is suggested, that is end-to-end trainable. Additionally, a decoder enhancement strategy for improved localization is introduced. The following content has been previously published in [Pha+19a] and has been revised for this section.

4.3.1 Methods

The general aim is to leverage ground truth information for a good segmentation prediction. This is usually already accomplished by comparing the network output with the ground truth by means of a loss function, that is minimized during the training phase. The key idea of this chapter is to distill relevant information about the structure of interest's shape from the ground truth information and incorporate this explicit information into the learning process of the neural network. Although the network is capable of learning shape aware features by using ground truth information by itself during training, it is not guaranteed that these features are actually learned for the segmentation task, as Geirhos et al.'s [Gei+18] demonstrate for image classification tasks. The explicit infusion of shape information can be considered as an additional emphasis to learn shape aware features.
For segmentation tasks, the ground truth is usually depicted as a simple label map, assigning each pixel the desired class label. Therefore, the information is sparsely distributed over a large tensor, i.e. a high dimensional data structure. For the compression of this unnecessary sparsity to a compact vector, convolutional autoencoders (CAEs) can be used as canonical deep learning methods.

Imitating the Encoder Component of the CAE

An interesting property of the CAE is that its decoder component is able to reconstruct the input with only little information loss from a compact representation of the input in latent space. When trained with ground truth maps, the decoder is therefore able to estimate the desired ground truth segmentation from a latent representation of the ground truth. Thus, if an image input can be compressed in a way, such that its representation is very similar to its corresponding ground truth compression, a robust decoder should be able to compute a segmentation prediction that is close to the desired ground truth segmentation. Motivated by Girdhar et al.'s TL-embedding network [GF+16], this generative property is used to obtain a feasible segmentation, given an arbitrary feature vector in latent space. Similar to Oktay et al.'s work [OF+18], the CAE is used to find an embedding in latent space, that encodes the anatomical priors, given by the ground truth.

For this purpose, an *imitating encoder* is proposed, that has the main goal of compressing an input image in a similar fashion as the CAE's encoder compresses the corresponding ground truth. This new encoder mimics or *imitates* the CAE's encoder. While the CAE's input and the desired output are the same, i.e. the ground truth map, the imitating encoder's input is the corresponding medical gray scale image or volume. Therefore, the imitating encoder basically reduces the input image to the important anatomical priors in latent space, in order to be reconstructed to a segmentation by the CAE's decoder component. In the following, an input image will be denoted as I and the segmentation ground truth as GT. $\Theta_{(\cdot)}$ represents the trainable weights of the architecture, where the subscript $_{(\cdot)}$ specifies, which parts of the architecture are considered. The CAE's encoder serves the purpose of encoding anatomical shape priors, therefore it will be referred to as a *prior encoder*, denoted as f_{enc_p}, whereas the generative decoder is referred to as g_{dec}. The mapping of GT into latent space is formalized as

$$\hat{z} = f_{enc_p}(GT, \Theta_{enc_p}),$$

whereas reconstruction from $\hat{z}$ to the CAE's output y_{CAE} is computed as

$$y_{CAE} := g_{dec}(\hat{z}, \Theta_{dec}) = g_{dec}(f_{enc_p}(GT, \Theta_{enc_p}), \Theta_{dec}).$$

The imitating encoder will be denoted as f_{enc_i}, therefore the imitation of $\hat{z}$ in latent space from I is

$$\hat{z} \approx \tilde{z} = f_{enc_i}(I, \Theta_{enc_i}).$$

The formulation of an imitation loss

$$\mathscr{L}_{imit}(\Theta_{enc_i}|\Theta_{enc_p}) := ||\hat{z} - \tilde{z}||_1, \tag{4.10}$$

where Θ_{enc_i} is adaptable and Θ_{enc_p} is fixed, enforces f_{enc_i} to encode the input in a similar fashion to f_{enc_p} during training. The idea is to utilize the CAE's decoder g_{dec} to achieve a segmentation from the input x, i.e.

$$\tilde{y}_{ie2d} =: g_{dec}(\tilde{z}, \Theta_{dec}) = g_{dec}(f_{enc_i}(I, \Theta_{enc_i}), \Theta_{dec}).$$

Enhancing the Localization Capability of the CAE's Decoder

A major disadvantage of this kind of encoder-decoder architecture for segmentation tasks is the loss of local information in the compressing encoder component, as pooling operations decrease the spatial size with the expense of positional information. Ronneberger et al. [RFB15] address this problem by using skip connections, introduced in Long et al.'s Fully Convolutional Networks [LSD15], from encoder to decoder, where the local information is preserved before any pooling operation. The decoder component g_{dec} of the CAE, however, cannot be equipped with skip connections from its own prior encoder f_{enc_p}, as these would be used to bypass the compression in latent space. A further counter argument against the usage of skip connections from f_{enc_p} to g_{dec} is that during inference ground truth information is not available, rendering f_{enc_p} obsolete. In order to use positional information for g_{dec} nonetheless, this information can be extracted from the input image. One possibility is to use the extracted features from the imitating encoder f_{enc_i} and pass them to g_{dec}. However, since the sole purpose of f_{enc_i} is to imitate f_{enc_p} only in latent space, it is not ensured that the learned features of f_{enc_i} are suitable for the main objective

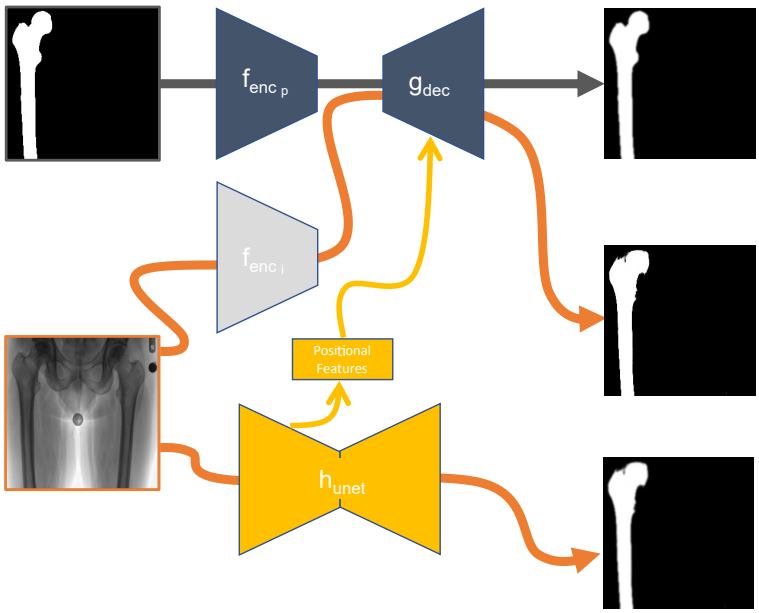


FIGURE 4.9: Schematic overview of IE$_2$D-Net. Shape information is compressed by the CAE, comprising f_{enc_p} and g_{dec}. Given a gray scale image, the imitating encoder f_{enc_i} tries to mimic f_{enc_p} in latent space. The U-Net module h_{unet} learns to provide hierarchical features for an enhanced decoding by g_{dec}.

of segmenting the input image. Alternatively, a U-Net module h_{unet} can be trained specifically for this purpose, as its contracting path learns to extract positional features, that are specifically relevant for segmentation tasks. Therefore, the decoder g_{dec} is enhanced by these additional learned hierarchical features from the U-Net module h_{unet} for an improved localization capability, altogether resulting in an *Imitating Encoder - Enhanced Decoder Network* (IE$_2$D-Net). A simplified overview of the architecture components is depicted in Fig. 4.9.

Training

The proposed architecture requires the ground truth information as input for its CAE component to learn a feasible latent representation, from which its decoder can reconstruct a segmentation prediction. This ground truth information is, however, usually not available during inference. Therefore, some components are omitted for inference. During training, however, all network components are considered. Moreover, multiple loss functions are defined, which are motivated as follows:

- The CAE component aims at learning a compact representation of the ground truth by minimizing a CAE loss term, comparing CAE's output y_{CAE} with the ground truth GT. This minimization process is restricted to only adapting a subset of all available network parameters, particularly Θ_{enc_p} and Θ_{dec}.
- Since the decoder component is enhanced with the hierarchical features from the U-Net component, these features are simultaneously learned by minimizing a U-Net output loss term, comparing the U-Net output, with the same ground truth, which is also fed into the CAE. This optimization is conducted by a separate optimizer, only focusing on Θ_{unet}, not inferring with the learning process of any other modules.

- At the same time the imitation property of the imitating encoder f_{enc_i} is enforced by using the imitation loss, described in Eq. (4.10). The optimization is again limited to the adaptation of the imitating encoder's weights Θ_{enc_i}, as its sole purpose is to encourage the encoder to mimic the feature representation in the CAE's latent space.
- To ensure a feasible prediction from the cascade of imitating encoder f_{enc_i} and generative decoder g_{dec}, a fourth IE$_2$D-Net output loss term is introduced, comparing the prediction from this cascade and the desired ground truth. Here, only the weights Θ_{enc_i} and Θ_{dec} are adapted.
- For g_{dec} the incoming skip connections are considered as constants.

A detailed overview of the architecture and its loss terms is depicted in Fig.4.10.

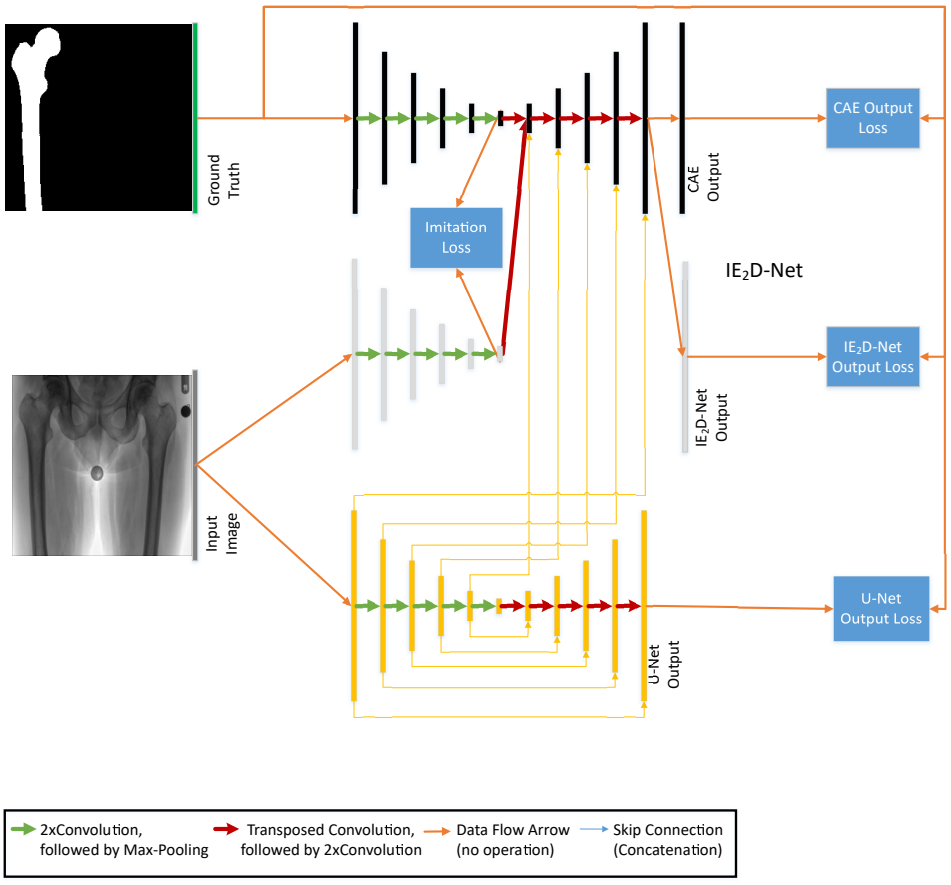


FIGURE 4.10: Detailed network architecture of IE$_2$D-Net. The architecture basically consists of three modules: the CAE module (top), the imitating encoder module (middle) and the U-Net module (bottom). For inference the prior encoder part of the CAE is omitted. Adapted from [Pha+19a], copyright ©2019 IEEE.

The network weights are adapted in every batch iteration by minimizing the aforementioned loss functions in the following order:

- First, the U-Net output loss is considered. The adaptation restriction to the U-Net weights ensures generation of hierarchical features in the contracting path, which are suitable for segmentation.

- Then, the CAE output loss is optimized. Again, the restriction ensures the CAE to find a meaningful prior representation in latent space, instead of depending on the features generated in the other modules.
- The third loss function to be minimized is the imitation loss.
- The last loss function consists of the IE$_2$D-Net output loss, i.e. a loss function that quantifies the segmentation quality of the combination of imitating encoder f_{enc_i} and the CAE's enhanced decoder g_{dec}.

Since the CAE module and the IE$_2$D-Net module share the same decoder component, the corresponding loss functions need to be minimized successively. The successive optimization is also applied for the IE$_2$D-Net output loss and the imitation loss, since both are dependent on the imitating encoder weights Θ_{enc_i}. A combination of imitation loss and IE$_2$D output loss is avoided to circumvent the necessity for an additional weighting scheme, which would also need to take into consideration that both losses are not in the same value range.
A major difference to related work lies in the end-to-end training approach in this proposal, in which both CAE and imitating encoder are trained at the same time, whereas the CAEs in Girdhar et al.'s and Oktay et al.'s approaches are pretrained.

Inference

During inference the encoding component of the CAE f_{enc_p} is replaced with the imitating encoder f_{enc_i}. Therefore, the U-Net module h_{unet} extracts relevant features in each scale level in its contracting path. At the same time the imitating encoder f_{enc_i} projects the input into a compact representation $\tilde{z}$ in latent space. This representation is then decoded by the decoder component g_{dec}, enhanced by the extracted U-Net features, resulting in the final segmentation prediction.

4.3.2 Experiments

The following experiments have been previously published in [Pha+19a] and show initial results of the designed architecture. Additionally, revised results are presented in this subsection. Any revisions and deviations from the original publication will be marked accordingly.

Data

For the experiments the same eight T1-weighted MR volumes of the pelvic regions are used as in chapter 3.3.5. In this case, hip bone annotations are used instead of femur annotations. Again, the data sets are denoted as P1, . . . ,P6 and the post operative data sets as P1PO and P2PO. The experiments are conducted in a leave-one-out cross validation manner, in which one data set is kept for testing, and the remaining data sets are used for training. In case of P1, P1PO, P2 and P2PO only those data sets are used for training, which do not correspond to the same patient. In the original publication P6 is used as an arbitrary validation patient to monitor the training process and is excluded from evaluation. For completeness P6 is also considered as a testing fold in this revised version.

Implementation Details

As baseline reference a variant of Ronneberger et al.'s 2D U-Net is implemented like in chapter 4.2.2. Again, two convolutional layers are used in each scale level and the

number of kernels doubles after each max pooling layer and halves after every transposed convolutional layer. The IE$_2$D-Net is designed in a similar fashion, following the blueprint of Fig. 4.10. The configuration details of the original publication and this revised version can be found in Table 4.3. For the imitation loss the euclidean distance is used and for the remaining loss functions the dice loss is employed. An Adam optimizer with an initial learning rate of 0.001 is applied for all experiments, which are conducted on a NVIDIA GTX 1080 ti GPU.

	original publication	**revision**
Framework	Tensorflow 1.12	PyTorch 1.10
Input size	128×128	256×256
Kernel size	10×10	3×3
Pooling size	2×2	2×2
Batch size	4	4
Epochs	40	100
U-Net levels	5	5
Augmentations	translation, rotation	translation, rotation, zoom

TABLE 4.3: Implementation details.

Results

Table 4.4 shows the achieved DSC values from the U-Net and the proposed IE$_2$D-Net in comparison for both the original publication and also the revision.

	original publication		revision	
	U-Net	**IE$_2$D-Net**	**U-Net**	**IE$_2$D-Net**
P1	0.6342	**0.6632**	0.8789	**0.8834**
P1PO	0.5719	**0.6715**	0.880	**0.9060**
P2	0.6784	**0.7466**	0.8212	**0.8377**
P2PO	0.7685	**0.8456**	**0.9464**	0.9456
P3	**0.8034**	0.7650	0.8006	**0.8606**
P4	0.7885	**0.7886**	0.8838	**0.9172**
P5	0.6525	**0.6613**	0.7439	**0.7650**
P6	-	-	0.8898	**0.9087**
$\emptyset$	0.6994±0.0744	**0.7345±0.0592**	0.8556±0.0635	**0.8780±0.0531**

TABLE 4.4: Left part: resulting average DSCs of the U-Net and IE$_2$D-Net for each patient from [Pha+19a], copyright ©2019 IEEE. Right part: average DSCs of revised experiments, including P6.

In both experimentation series, the proposed IE$_2$D architecture yields superior results, compared to the U-Net predictions. For all cases in which the U-Net shows better performance, the IE$_2$D approach yields at least close DSC values, which is not observable for the reversed case.
In [Pha+19a] an average DSC of 0.7345 ± 0.0593 compared to an average DSC of 0.6994 ± 0.0744 for the U-Net is achieved. In this revision, the overall results improve, presumable because of the increased number of epochs and augmentations, and the larger input size, allowing the extraction of more detailed structures. Here the obtained average DSC of IE$_2$D-net is 0.8780 ± 0.0531 compared to an average DSC

of 0.8556 ± 0.0635 for the U-Net baseline.

sHD	U-Net	IE$_2$D-Net
P1	48.867168	**41.412560**
P1PO	61.131008	**46.400433**
P2	84.952927	**61.619801**
P2PO	**88.300621**	149.17104
P3	56.044624	**54.046276**
P4	**25.337719**	41.617306
P5	**63.198101**	64.288414
P6	**9.4339809**	53.037724
$\emptyset$	**54.6583**$\pm$25.2606	63.9492$\pm$33.1663

TABLE 4.5: sHDs of revised experiments including P6.

Table 4.5 shows the measured symmetric Hausdorff distances (sHD) of the resulting predictions for each test patient. Interestingly, the sHD does not reflect the achieved DSC in Table 4.4. Here, U-Net's segmentation predictions yield an average sHD of 54.6583 compared to IE$_2$D-Net's sHD of 63.9492. A possible explanation for the poorer performance regarding sHD may be the changing shape of the pelvic bone along the axial slices. This may lead to outlier predictions for shapes, that the imitating encoder is not able to handle because of a limited representative capacity.

Fig. 4.11 depicts exemplary MR slices from four different patients, in which IE$_2$D-Net performs well. The desired ground truth segmentation is drawn in green, the U-Net output in red, and the IE$_2$D-Net results in blue. While the U-Net apparently has difficulties to encapsulate the bone contour, the proposed IE$_2$D-Net approach shows superior results in these examples. Especially in the bottom two samples, IE$_2$D-Net shows its shape preserving property (at least for the shape in this slice).

Fig. 4.12 on the other hand shows exemplary MR slices from four patients, in which the limitations of IE$_2$D-Net are illustrated. In the first row, both U-Net and IE$_2$D-Net confuse the femoral bone as a pelvic bone. Furthermore, an outlier prediction of IE$_2$D-Net is noticeable, that indicates the high sHD. In the second and third row, both architectures fail to encapsulate the whole pelvic bone. The last row shows an example, in which the shape preserving property is again disadvantageous for this setting, as the IE$_2$D-Net fails to follow the hole of the hip joint, which the U-Net is capable of. As mentioned before, the changing shape of the pelvic bone along the axial slices seems to be disadvantageous, if the model does not have enough representative capacity to store all shape information.

4.3.3 Summary

In this section, the IE$_2$D-Net is introduced as an end-to-end deep learning architecture, that realizes the infusion of shape priors by means of mimicking a CAE's encoder. The evaluation on the example of pelvic bone segmentation shows promising results, compared to the U-Net in both original publication and revised experiments regarding DSC. Although the IE$_2$D-Net results are significantly better (with a significance level of $p = 0.0156$, using Wilcoxon's signed rank test [Wil92]), the number of samples is, nevertheless, very small. The improvement's significance would be

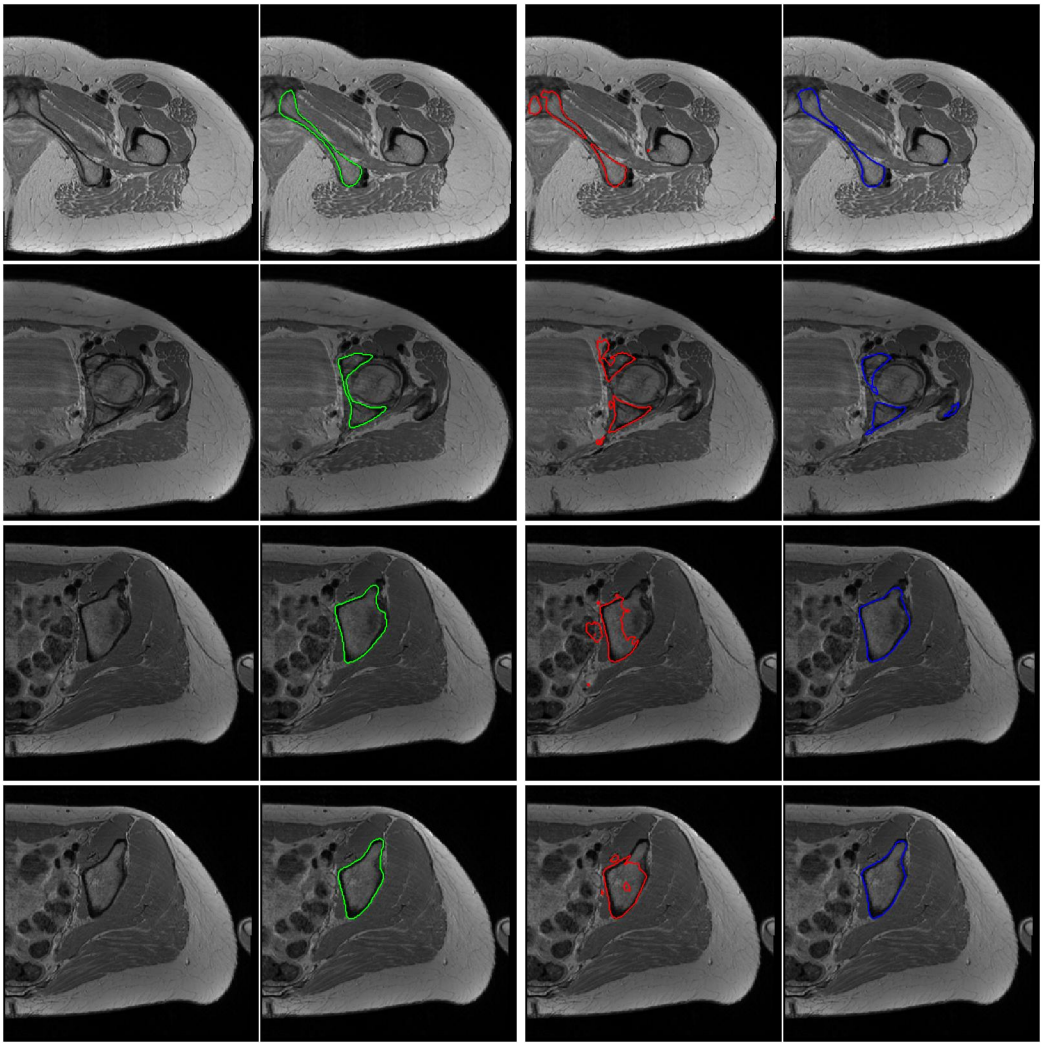


FIGURE 4.11: Exemplary good results. The first column shows the exemplary axial MRI slice. The ground truth (green) is depicted in the second column, the U-Net output (red) in the third, and the IE_2D-Net output (blue) in the last column.

more convincing for a larger test set. Furthermore, the application on volumes in a slice-wise manner may limit the architectures potential, as shapes of the same structure show too much variation along the slices. This is reflected in the poorer sHD performance compared to the U-Net baseline, even if the differences are not significant (significance level of $p = 0.8438$). Nevertheless, the performance on structures with less shape variation needs to be additionally investigated. This is done on the example of 2D femur extraction from fluoroscopic x-ray images in section 4.4.5 in the context of ablation studies of the extension **IRE_3D-Net**, proposed in the next section.

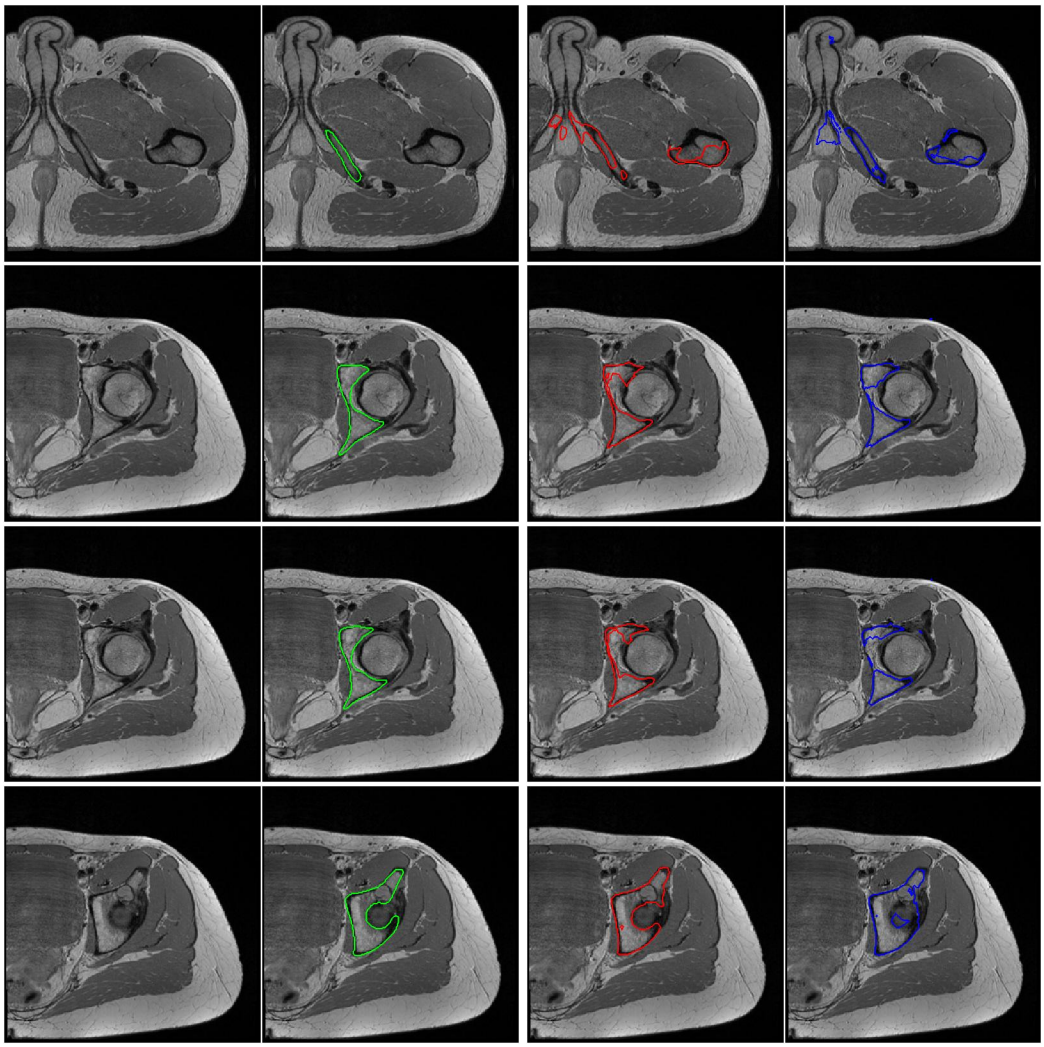


FIGURE 4.12: Exemplary bad results. The first column shows the exemplary axial MRI slice. The ground truth (green) is depicted in the second column, the U-Net output (red) in the third, and the IE_2D-Net output (blue) in the last column.

4.4 Imitating and Regularizing Encoders - Enhanced Decoder Network

Oktay et al.'s work [OF+18] motivates to extend the IE$_2$D-Net from the previous section 4.3 to reuse the already trained components for an additional enforcement of shape preservation. In this section, an *Imitating and Regularizing Encoders and Enhanced Decoder Network (IRE$_3$D-Net)* is presented, that is derived from the IE$_2$D-Net, additionally leveraging Oktay et al.'s enforcement of stronger shape regularization. The design and experiments have been previously published in [Kav+21] and [PDP21b]. In the first publication, the contribution mainly consists of the architecture description, implementation, and submission.

4.4.1 Methods

Analogously to the IE$_2$D-Net, described in section 4.3.1, the IRE$_3$D-Net consists of two convolutional encoders f_{enc_p}, f_{enc_i}, one decoder g_{dec} and one U-Net component h_{unet}. While f_{enc_p} and g_{dec} form a convolutional autoencoder (CAE), f_{enc_i} and g_{dec} constitute a segmentation hourglass network. The U-Net module h_{unet} is used to enhance g_{dec} for an image guided decoding process to increase the decoder's localization capabilities. However, the extended architecture design shows the following differences to the IE$_2$D-Net:

1. The CAE's prior encoder f_{enc_p} additionally serves as a regularization module, that measures the output's shape consistency in latent space during training, following Oktay et al.'s proposition [OF+18].

2. The prediction of the U-Net module h_{unet} is fused with the decoder prediction $g_{dec}(f_{enc_i}(I))$ from an input image I, by averaging their pixel-wise predictions, as depicted in Fig. 4.13 by a plus symbol.

3. Regularization terms to ensure shape consistency are added to the segmentation loss functions of the U-Net and IRE$_3$D outputs. The shape consistency regularization terms for the corresponding modules are defined as

$$\mathscr{L}_{SC_{ire3d}} := ||f_{enc_p}(g_{dec}(f_{enc_i}(I))) - f_{enc_p}(\hat{y})||_1,$$
$$\mathscr{L}_{SC_{unet}} := ||f_{enc_p}(h_{unet}(I))) - f_{enc_p}(\hat{y})||_1,$$

   for an input image I and the desired target segmentation ground truth $\hat{y}$. The regularization terms are further scaled with a weighting factor $\lambda_{reg} > 0$.

The overall architecture is depicted in Fig. 4.13. Otherwise the network is trained and used in the same manner as the IE$_2$D-Net.

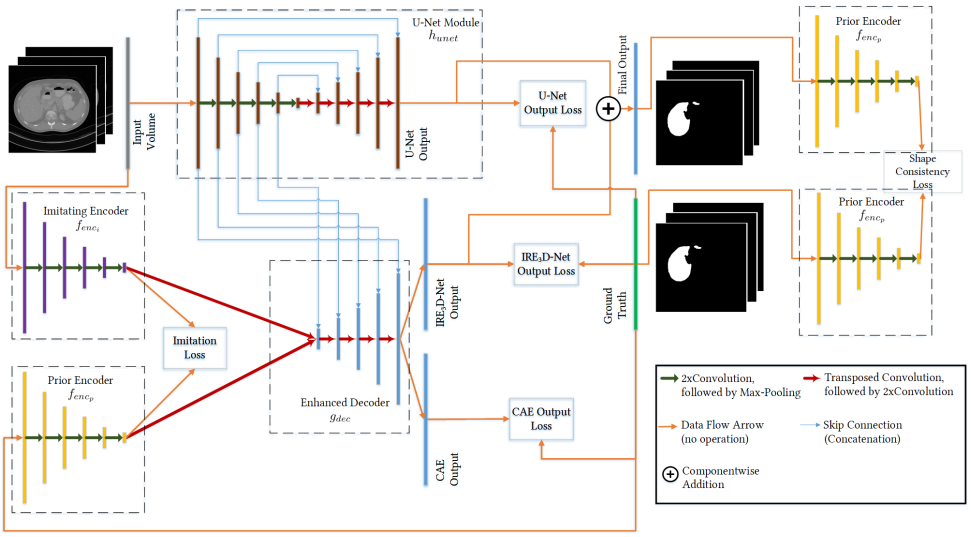


FIGURE 4.13: IRE$_3$D-Net architecture. The architecture consists of four modules: the U-Net module (brown), the imitating encoder (purple), the prior encoder (yellow), and the joint decoder (blue). The prior encoder is additionally used for shape consistency of the output segmentation and the ground truth, as described in point 1. Also, the U-Net module output is fused with the IRE$_3$D-Net output, as mentioned in point 2. The consideration of the shape consistency term in the segmentation losses (point 3) is, however, not visualized in this figure. For inference the prior encoder is omitted. Reused from [PDP21b].

4.4.2 Experimental Overview

The IRE$_3$-Net is applied in three different experimental settings. The first experiments deal with cross-modality organ segmentation, in which IRE$_3$D-Net is trained for abdominal organ segmentation with both MR and CT in different combinations, which will be shortly elaborated in the next but one subsection. This experimental setting is determined by the *Combined Healthy Abdominal Organ Segmentation (CHAOS)* challenge [Kav+21], held at the International Symposium on Biomedical Imaging (ISBI) 2019 in Venice, Italy. The evaluation is conducted in a hold-out validation manner, i.e. the labeled training data is available for training, whereas the test data labels are kept hidden by the organizers. In this subsection, the challenge design is only shortly outlined and the performance of the IRE$_3$D-Net is presented in this context. Further details can be found in Kavur et al.'s publication [Kav+21].

The second series of experiments investigates the applicability of the IRE$_3$D-Net in a one-shot segmentation setting. In this scenario, comparisons to IE$_2$D-Net and Oktay et al.'s Anatomically Constrained Neural Network (ACNN) [OF+18] are drawn on the example of liver segmentation in CT volumes. Here, a leave-one-out cross validation design is chosen to evaluate the network performances.

In the last series of experiments, ablation studies on the IRE$_3$D architecture are conducted in a regular segmentation setting, without any one-shot or cross-modality learning conditions. The experiments are performed on the example of single sided femur extraction in fluoroscopic x-ray images.

Implementation Details

For the first two experimental settings, the networks are designed to process 3D volumes. Therefore, a variant of Çiçek et al.'s 3D U-Net [ÇA+16] is implemented in a similar manner as the 2D U-Net design in sections 4.2.2 and 4.3.2. Two convolutional layers are used in each scale level and the number of kernels doubles after each max-pooling layer and halves after every transposed convolutional layer. The IRE_3D-Net is implemented accordingly as depicted in Fig. 4.13, as well as IE_2D-Net and ACNN. For the last setting, 2D architectures are implemented in the same manner. The net-

	Cross-modality organ segmentation	**One-shot segmentation**	**2D femur extraction**
Framework	Tensorflow 1.12	Tensorflow 1.12	PyTorch 1.10
Input size	$128 \times 128 \times 96$	$128 \times 128 \times 96$	256×256
Kernel size	$3 \times 3 \times 3$	$3 \times 3 \times 3$	3×3
Pooling size	$2 \times 2 \times 2$	$2 \times 2 \times 2$	2×2
Batch size	1	1	10
Epochs	40	2400	60
Pooling layers	5	5	5
Augmentations	translation, rotation	translation, rotation	translation, rotation, zoom

TABLE 4.6: Implementation details.

work configuration details for all experimental settings can be found in Tab. 4.6. An Adam optimizer with an initial learning rate of 0.001 is applied for all experiments. The weight of the regularization term in both the ACNN and the IRE_3D is set to $\lambda_{reg} = 0.001$, as suggested by Oktay et al. [OF+18]. All experiments are conducted on a NVIDIA GTX 1080 ti GPU.

4.4.3 Experiments 1: Cross-Modality Organ Segmentation

The CHAOS challenge aims at comparing different strategies for the segmentation of healthy abdominal organs, i.e. liver, spleen, and both kidneys. In particular, it comprises five sub-tasks to investigate the performance on various cross-modality settings:

1. Liver Segmentation (CT+MR)
   The training and testing sets comprise CT and MR data. Both training sets have the corresponding liver segmentation ground truths.

2. Liver Segmentation (CT only)
   The training and testing sets only contain CT data. The training set includes the corresponding liver segmentation ground truths.

3. Liver Segmentation (MR only)
   The training and testing sets only contain MR data. The training set includes the corresponding liver segmentation ground truths.

4. Segmentation of abdominal organs (CT+MR)
   The training and testing sets comprise both CT and MR data. The MR set includes the corresponding segmentation ground truths for all abdominal organs, whereas the CT set only has the segmentation ground truth for the liver.

5. Segmentation of abdominal organs (MR only)
   The training and testing sets only contain MR data. The training set includes the corresponding segmentation ground truth for all abdominal organs.

While tasks 2, 3 and 5 are single modality problems, tasks 1 and 4 address the concept of cross-modality learning.

Data

The CT data set consists of 40 volumes, from which 20 are provided with liver annotations for training. The remaining 20 volumes are used for evaluation, where the annotations are kept by the competition organizers. The MR data set on the other hand consists of 120 volumes from 40 patients, which are different from the patients of the CT data set. Here, each patient is scanned with two pulse sequences (T1 and T2), resulting in 40 T1-DUAL scans, and 40 T2-SPIR scans. The T1-DUAL scans comprise in-phase and out-of-phase representations, accumulating in a total number of 120 MR volumes including the T2 scans. From these 120 volumes only 60 are provided for training with corresponding labels, whereas the remaining ground truths are kept by the organizers.

Results

In the scope of this challenge, a scoring system is used, considering the DSC, the symmetric Hausdorff distance, both as described in section 2.4, the relative absolute volume difference (RAVD), and additionally the average symmetric surface distance, i.e. the average over all minimal distances from prediction boundary points to ground truth boundary and vice versa. The RAVD for a segmentation prediction volume $y^{(c)}$ and its corresponding desired ground truth volume $GT^{(c)}$ for any class $c \in \mathscr{C}$ is defined as

$$RAVD(y^{(c)}, GT^{(c)}) := \frac{||y^{(c)}| - |GT^{(c)}||}{|GT^{(c)}|} \cdot 100,$$

where $|\cdot|$ measures the magnitude of a volume, i.e. the number of all non-zero elements. To combine these four metrics, they are mapped to a range of $[0, 100]$, where a higher score represents better performance.

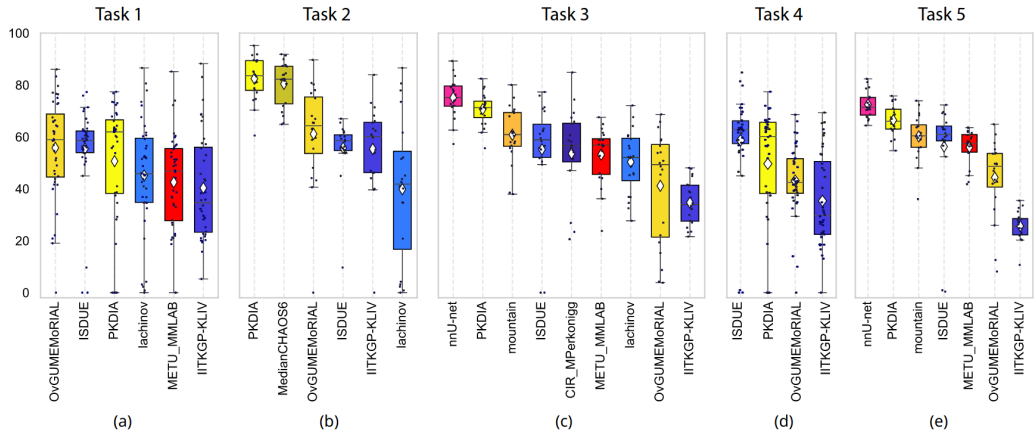


FIGURE 4.14: CHAOS results for each task. The proposed IRE$_3$D architecture is denoted as *ISDUE*. White diamonds represent the mean values of the scores and solid horizontal lines inside of the boxes represent the medians. Separate dots show scores of each individual test volume. This figure is taken from [Kav+21] with permission from both the publisher and Kavur, who evaluated the submissions and created this diagram.

Fig. 4.14 shows the performance results of the submitted designs, which are ranked from best to worst performance from left to right for each task, respectively. The proposed IRE$_3$D architecture is denoted as *ISDUE*, representing the submission of the **I**ntelligent **S**ystems group of the University of **Du**isburg-**E**ssen.
It is noticeable, that in the single modality tasks 3 and 5, using only MR data, the proposed IRE$_3$D architecture is outperformed by *nnU-net*, *PKDIA*, and *mountain*. While Isensee et al.'s *nnU-net* [IK+18; Ise+21] is a self-configuring U-Net pipeline, which adapts its configuration based on the underlying task, Conze et al.'s *PKDIA* submission [Con+21] uses an additional discriminator to critic their generator's segmentation prediction. Here, the generator is represented by a 2D U-Net implementation. The *mountain* team makes use of a cascade of U-Net variants, in which residual blocks are incorporated in the contracting path. Furthermore, strided convolutions are used instead of max poolings.

A similar observation can be made for the single modality task 2, using only CT data, as *PKDIA*, *MedianCHAOS6*, and *OvGUMEMoRIAL* yield better results than the proposed IRE$_3$D-Net. *MedianCHAOS6* is the 6-th submission of an ensemble method, using multiple U-Net variants, and *OvGUMEMoRIAL* makes use of a U-Net variant. It stands out, that Ronneberger et al.'s U-Net is the base architecture, that all well performing submissions in these single modality tasks build on.

In the cross-modality tasks 1 and 4, the proposed IRE$_3$D-net achieves favorable scores, also showing only little variance in performance across the test set. Therefore, the inclusion of shape information appears to be advantageous in cross-modality settings, particularly when the same structure needs to be extracted from both modalities as in task 1. For task 4, the imitation strategy may be beneficial, as this mechanism allows the differentiation between multi-organ and single-organ segmentation. Since ground truth maps for a single organ are more sparse than ground truth maps for multiple organs, their latent representations in the CAE module certainly differ significantly from the latent multi-organ representations. Because of the imitation mechanism, CT scans are projected towards single-organ representations, whereas MR volumes are mapped to multi-organ representations, from which the CAE's decoder constructs the prediction.

Nevertheless, it needs to be kept in mind, that different training capacities and the absence of architectural restrictions renders direct comparison more difficult. For instance, *nnU-net* and *MedianCHAOS6* make use of ensemble methods, which often lead to better predictions. However, *nnU-net* is only submitted once, whereas *MedianCHAOS6* is tuned towards task 2 in multiple submissions. Regarding training capacity, *mountain* is able to train on volumes of size $(384 \times 384 \times 64)$, whereas *PKDIA* circumvent the GPU size limitation by using a 2D slice-wise approach to leverage the higher resolution in 2D. Since shapes may vary significantly in a slice-wise approach (see section 4.3.2), the proposed IRE$_3$D-Net is trained on volumes of size $(128 \times 128 \times 96)$.
Altogether, the challenge shows, that IRE$_3$D-Net yields competitive results, especially for cross-domain learning tasks, although it is trained on resized volumes of smaller size.

4.4.4 Experiments 2: One-Shot Image Segmentation

The second series of experiments investigates the applicability of IRE_3D-Net in the context of one-shot segmentation, in which the network architecture only has access to one sample during training, trying to generalize to unseen samples (of the same modality). This part particularly analyses the effects of additional shape priors in the generalization performance, compared to a 3D U-Net. As reference architectures, that also consider shape information, Oktay et al.'s anatomically constrained neural network (ACNN) and the IE_2D-Net are considered. In this scope, the ACNN uses a 3D U-Net as its base segmentation network and the IE_2D-Net is implemented to handle 3D inputs. The performance is measured by means of the Dice Similarity Coefficient (DSC) and the symmetric Hausdorff distance (sHD).

Data

The one-shot experiments are conducted on the example of liver segmentation from abdominal CT volumes. For this, data from the Cancer Imaging Archive [CV+13; RF+16; RL+15] (TCIA) and from the Beyond the Cranial Vault (BTCV) segmentation challenge [LX+15; XL+16] is used. The supplementary ground truth segmentations are published by Gibson et al. [GG+18]. In total 90 abdominal CT volumes with corresponding ground truths are used.

In a leave-one-out cross validation manner, every architecture is trained 90 times, using each patient data set as one-shot training set once. Patients 40 and 90 are arbitrarily used as validation data sets. When training with patient 40, patients 39 and 90 are used for validation. When training with patient 90, patients 40 and 89 are used for validation. The remaining patient volumes, that have not been involved in training or validation, are then used for testing.

As a reference (Ref) the DSC, which is achieved by just regarding the ground truth segmentation of the training data set as segmentation prediction for each test volume, is additionally calculated. Furthermore, a *situs inversus* case is simulated, in which the organs are flipped. This is motivated by the intention to investigate how the trained networks react to cases that topographically differ from physiological images. The situs inversus case is simulated by inverting the stack ordering in longitudinal axis for validation and all test data sets, while keeping the ordering for the training set.

Results

Since the complete data set is taken from two different sources, one can differentiate between 4 cases regarding evaluation results:

- Q_{11}: Trained on TCIA and tested on TCIA
- Q_{12}: Trained on TCIA and tested on BTCV
- Q_{21}: Trained on BTCV and tested on TCIA
- Q_{22}: Trained on BTCV and tested on BTCV

These four cases are especially noticeable for the reference DSC measures in Fig.4.15, where the achieved DSCs for each train/test patient combination is depicted as a heat map. Rows indicate the patient data set used for training, whereas columns

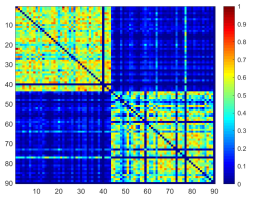


FIGURE 4.15: DSC Heatmap of reference for ordinary setting. Rows indicate patient used for training, columns represent test data set. Reused from [PDP21b].

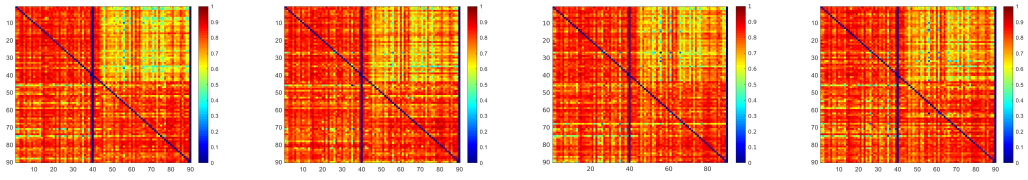
represent the patient test data set. A strong DSC change is visible from column 43 to 44 and row 43 to 44, dividing the heat map into the four quarters Q_{11}, Q_{12}, Q_{21}, and Q_{22}. In particular, the DSCs get worse, when ground truths come from different data sources, i.e. in the upper right quarter Q_{12} and the lower left Q_{21}. This indicates that there already is a strong overlap between ground truths within each data source. The segmentation problem might therefore be easier when train and test patients come from the same data source.

This hypothesis is supported by the segmentation results of the considered architectures. Table 4.7 shows the resulting mean DSCs for each architecture in each quarter and the overall DSCs. For all considered architectures, the best DSCs are achieved in

DSC	Q_{11}	Q_{12}	Q_{21}	Q_{22}	$\emptyset$
Ref	0.537 ± 0.12	0.317 ± 0.13	0.317 ± 0.13	0.407 ± 0.20	0.382 ± 0.18
U-Net	**0.829** ± 0.06	0.673 ± 0.11	0.784 ± 0.09	**0.803** ± 0.07	0.771 ± 0.10
ACNN	0.825 ± 0.06	0.696 ± 0.10	**0.790** ± 0.08	0.797 ± 0.08	0.776 ± 0.10
IE_2D	0.821 ± 0.07	0.702 ± 0.11	0.784 ± 0.10	0.803 ± 0.07	**0.777** ± 0.10
IRE_3D	0.818 ± 0.07	**0.705** ± 0.11	0.776 ± 0.10	0.801 ± 0.08	0.774 ± 0.10

TABLE 4.7: Achieved mean DSCs in each case. Reused from [PDP21b].

Q_{11} and Q_{22}. The standard U-Net yields the best results regarding DSC in Q_{11} and Q_{22}, while architectures with anatomical priors show slightly better results in Q_{12} and Q_{21}, i.e. when data sources for training and testing are different. This section's IRE_3D architecture achieves the best DSC result in Q_{12}, whereas for Q_{21} Oktay et al.'s ACNN surpasses the remaining models. Regarding mean DSC over all test cases, all architectures with anatomical priors show slightly better results than U-Net.
Since the improvements are only marginal, significance tests are conducted. As there are 8100 test cases, *two-sample t-tests*, introduced by Gosser under the pseudonym *Student* [Stu08], are more suitable than Wilcoxon's signed rank test [Wil92]. The t-tests yield that only ACNN and IE_2D-Net significantly improve the overall DSC compared to U-Net with p-values of 0.0025 and $8.6814e-04$, respectively. As can be seen in Q_{11} and Q_{22}, U-Net outperforms the proposed models when training and testing data come from the same source. Therefore a possible explanation for IRE_3D-Net's poorer overall segmentation results compared to ACNN and IE_2D-Net regarding DSC lies in the aggregation of limitations of ACNN and IE_2D-Net in these scenarios. This is because IRE_3D-Net can be considered a combination of ACNN and IE_2D-Net. However, it needs to be emphasized that these observations are still marginal and can only be seen in the context of DSC in one-shot segmentation settings.



(a) U-Net (b) ACNN (c) IE_2D-Net (d) IRE_3D-Net

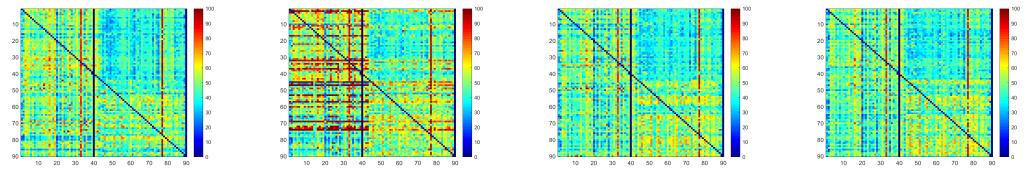
FIGURE 4.16: DSC heatmaps for each train/test combination of U-Net, ACNN, IE_2D-Net, IRE_3D-Net. Rows indicate patients used for training, columns for testing. Red indicates high scores, blue implies low values. Reused from [PDP21b].

Fig.4.16 depicts the DSC heatmaps of all trained models. Here it becomes visually apparent, that the transition from TCIA to BTCV (Q_{12}), seems to be especially difficult for all models, as the upper right quadrant shows lower DSCs than the remaining quadrants for all models.

sHD	Q_{11}	Q_{12}	Q_{21}	Q_{22}	$\emptyset$
U-Net	50.8 ± 12.9	**40.4**$\pm$11.1	48.6 ± 14.2	**48.2**$\pm$12.7	50.3 $\pm$15.4
ACNN	60.1 ± 19.1	45.6 ± 13.3	55.7 ± 23.5	54.8 ± 14.1	58.3 $\pm$20.5
IE_2D	48.1 ± 13.4	40.5 ± 10.5	45.9 ± 13.1	51.7 ± 12.4	48.6 $\pm$14.1
IRE_3D	**47.8**$\pm$12.8	40.4 ± 10.7	**45.1**$\pm$12.8	51.1 ± 12.8	**48.5** $\pm$14.5

TABLE 4.8: Mean sHDs in each case. Reused from [PDP21b].

The observation that models with shape priors perform better when source and target domain are different is, however, only partly reflected by the Hausdorff distances in Table 4.8. Surprisingly, the regularization in the ACNN results in higher sHDs than for the standard U-Net in all cases. The IRE_3D-Net shows the best results regarding sHD in Q_{11} and Q_{21}, whereas in the challenging Q_{12} case it performs equally well as the U-Net.
The IRE_3D-Net shows the best mean sHD performance over all test cases, followed by IE_2D-Net. Both architectures significantly improve the mean sHD with p-values of $1.0431e-13$ and $5.1020e-14$, respectively. A significant difference between achieved mean DSC and sHD of IE_2D-Net and IRE_3D-Net could, however, not be shown, at least not for one-shot segmentation settings.



(a) U-Net (b) ACNN (c) IE_2D-Net (d) IRE_3D-Net

FIGURE 4.17: Hausdorff distance heatmaps for each train/test combination of U-Net, ACNN, IE_2D, and IRE_3D (from left to right). Rows indicate patients used for training, columns for testing. Red indicates high distances, blue implies low values. Reused from [PDP21b].

Fig. 4.17 shows the heat maps of measured sHDs for each train/test scenario in each model. Here the surprising observation is visualized, as it can be seen that the

ACNN architecture seems to have difficulties especially in Q_{11} and Q_{21}. While the models with shape prior information show promising results for one-shot settings, in which source and target domain are different, it is still surprising that a standard U-Net is also capable of achieving similarly good DSC and sHD scores and even better scores in Q_{11} and Q_{22}, when only trained with one patient volume.
The next paragraph will investigate if this observation still holds, when train and test data show a stronger deviation, such as in the simulated situs inversus setting.

Situs Inversus Simulation

Situs inversus is a very rare condition, in which the inner organ positions are mirrored along the vertical axis. Therefore the positions are inverse to the physiological position (situs solitus). The situs inversus case is simulated by inverting the stack ordering in longitudinal axis for validation and all test data sets, while keeping the ordering for the training set.

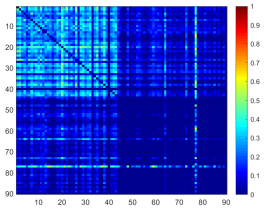


FIGURE 4.18: DSC heatmap of reference for situs inversus setting . Rows indicate patients used for training, columns represent test data. Reused from [PDP21b].

The reference DSC heat map in Fig. 4.18 shows that Q_{22} yields the most challenging case, whereas Q_{11} seems to be the easiest scenario. This may be due to the fact, that in BTCV the liver is not centered along the longitudinal axis, because of the wider field of view. Thus, when inverting the stack order, the overlap of liver regions between training and testing data set is smaller than in the other cases. In TCIA, particularly, the liver is more centered, such that even when inverting the order, the overlap of liver regions is still rather large. This circumstance can be seen in Fig. 4.19, where the original CT volumes and the re-ordered volumes are depicted next to each other in a frontal view for TCIA and BTCV, respectively.

	Q_{11}	Q_{12}	Q_{21}	Q_{22}	$\emptyset$
Ref	0.260 ± 0.10	0.083 ± 0.08	0.083 ± 0.08	0.014 ± 0.05	0.102 ± 0.12
U-Net	**0.762**$\pm$0.07	0.593 ± 0.11	0.604 ± 0.17	0.450 ± 0.18	0.595 ± 0.18
ACNN	0.760 ± 0.07	0.611 ± 0.11	0.597 ± 0.19	0.459 ± 0.18	0.599 ± 0.18
IE_2D	0.754 ± 0.08	0.618 ± 0.11	**0.637**$\pm$0.16	**0.524**$\pm$0.16	**0.628** $\pm$0.15
IRE_3D	0.758 ± 0.08	**0.630**$\pm$0.10	0.619 ± 0.16	0.512 ± 0.16	0.624 ± 0.16

TABLE 4.9: DSCs for situs inversus setting. Taken from [PDP21b].

Table 4.9 reveals, that both the IE_2D architecture and the IRE_3D-Net outperform the standard U-Net in all cases except for Q_{11}, where the standard U-Net yields best DSC results. The best mean DSC over all test cases is achieved by IE_2D-Net, followed by IRE_3D-Net and then ACNN. The improvements for ACNN could not be established

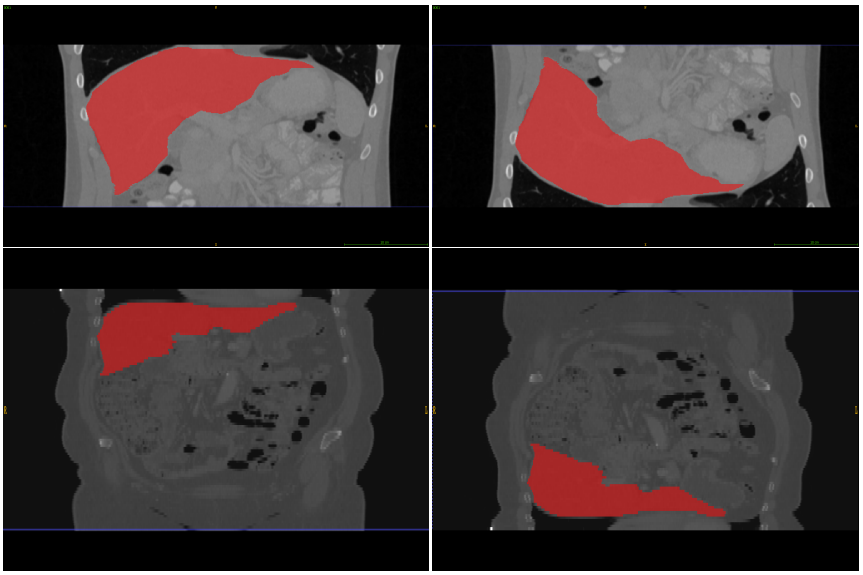
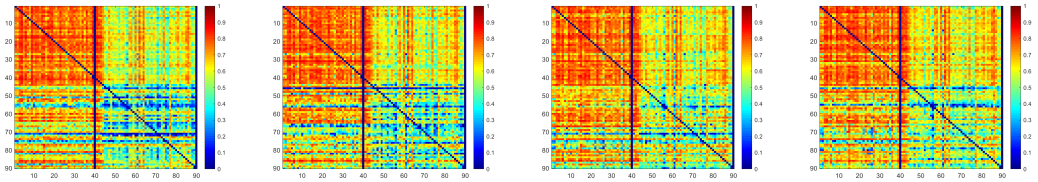


FIGURE 4.19: Comparison of liver regions for situs inversus case. The top row shows an example from the TCIA data set, which is reordered, and the bottom row an example from the BTCV data set. It is directly visible, that the liver region overlap for TCIA is larger than for BTCV, where the field of view is larger.

as significant. However, IE_2D-Net and IRE_3D-Net yield significant improvements compared to U-Net with p-values of $3.2423e-35$ and $1.3575e-26$, respectively.



(a) U-Net (b) ACNN (c) IE_2D-Net (d) IRE_3D-Net

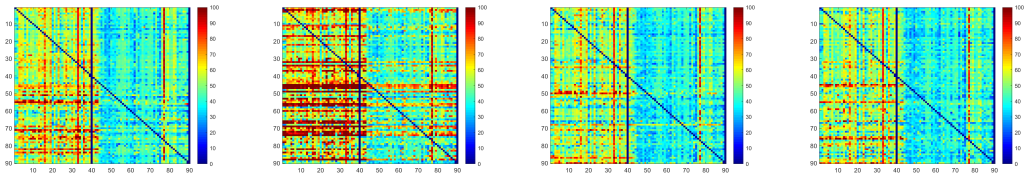
FIGURE 4.20: DSC heatmaps for each train/test combination of U-Net, ACNN, IE_2D, and IRE_3D for the situs inversus setting. Rows indicate patients used for training, columns represent the test volume. Reused from [PDP21b].

Figure 4.20 shows the DSC heatmaps of the trained models for each train/test combination. It is directly noticeable that for Q_{11} all models perform very well, whereas the most problematic case is Q_{22}, i.e. when there is only little liver region overlap between training and testing image.

sHD	Q_{11}	Q_{12}	Q_{21}	Q_{22}	$\emptyset$
U-Net	55.7 ±12.1	42.0 ±11.3	60.5 ±17.8	42.0 ±11.2	46.3 ±13.5
ACNN	64.4 ±18.0	46.9 ±14.0	71.9 ±24.4	51.6 ±18.0	54.0 ±19.7
IE_2D	52.3 ±12.8	40.1 ±10.6	**54.2**±15.5	39.6 ±10.6	44.3 ±12.8
IRE_3D	**51.4** ±12.3	**39.8** ±10.4	55. 2±16.6	**39.2**±10.3	**43.7** ±12.4

TABLE 4.10: sHDs in situs inversus setting. Taken from [PDP21b].

Considering the Hausdorff distances in Table 4.10, IE_2D and IRE_3D also show better results than the U-Net in all cases. It is, however, surprising that the shape regularization in ACNN, again, results in considerably worse Hausdorff distances in all cases.
Regarding overall sHD, IRE_3D-Net achieves the best results, followed by IE_2D-Net and U-Net. The improvements of IRE_3D-Net and IE_2D-Net are significant with p-values of $1.4702e-22$ and $2.7155e-35$, respectively. The difference of IRE_3D-Net and IE_2D-Net could also be established as significant with a p-value of 0.0096.



(a) U-Net (b) ACNN (c) IE_2D-Net (d) IRE_3D-Net

FIGURE 4.21: Hausdorff distance heatmaps for each train/test combination of U-Net, ACNN, IE_2D, and IRE_3D (from left to right) for situs inversus setting. Rows indicate patients used for training, columns represent test volume. Reused from [PDP21b].

These observations are visualized in Fig.4.21, in which the worse Hausdorff distances of the ACNN are noticeable in the larger amount of red heat map positions. It is surprising that U-Net seems to be superior to ACNN in most cases regarding Hausdorff distance. For IE_2D and IRE_3D, however, Fig.4.21 underlines that in most cases U-net achieves inferior distances.

Figure 4.22 depicts exemplary segmentation results for the situs inversus setting. In the first six examples, the worse sHD of ACNN become immediately apparent, as there seem to be outliers, following a specific pattern. If these outliers are overlooked, the examples particularly show, that the incorporation of Oktay et al.'s regularization scheme in general yields smoother surfaces on both the ACNN and IRE_3D-Net results, which is reflected by the measured DSC scores. The IRE_3D-Net also produces the least amount of outliers, which is also implied by the significantly lower sHDs.

Altogether, these experiments demonstrate, that the incorporation of shape priors yields improved segmentation results, especially in cases where the organ of interest does not strongly overlap between training and test volume, as can be particularly seen in the situs inversus setting.
On the other hand, U-net surprisingly already performs well in one-shot segmentation settings, if the organ location strongly overlaps in training and test volume. Moreover, IE_2D- and IRE_3D-Net show improved but also very similar results. In the one-shot situs inversus setting, the extension of IE_2D-Net to IRE_3D-Net is able to significantly improve the overall sHD. It is however questionable, whether this observation also holds in regular segmentation settings without any one-shot or cross-modality learning conditions. This is investigated in the following series of experiments of the next subsection.

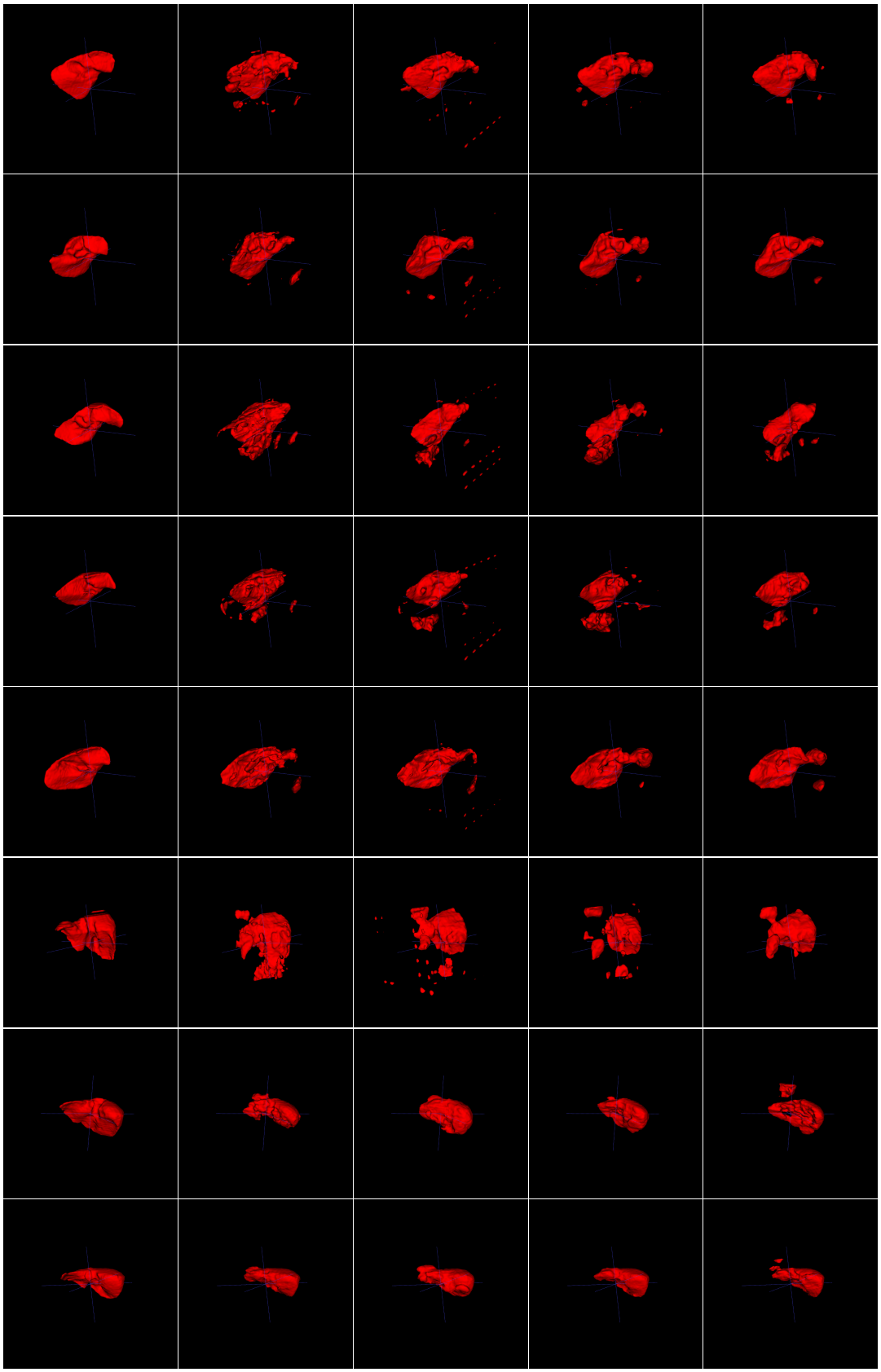


FIGURE 4.22: Exemplary one-shot segmentations from each architecture.

4.4.5 Experiments 3: Ablation Studies

The third series of experiments conducts ablation studies on the proposed IRE$_3$D architecture on the example of femur extraction in fluoroscopic x-ray images. While the IRE$_3$D design shows promising results in the cases of cross-modality training and one-shot segmentation, the necessity of the additional modules in the context of regular image segmentation still needs to be investigated.
Therefore, each component is removed from the proposed architecture, to possibly achieve a slimmer network design. Removing the additional regularizing encoder from the *IRE$_3$D-Net* results in the *IE$_2$D-Net*. Dismissing the U-Net module from the IE$_2$D-Net, results in an architecture denoted as *IED-Net*, where positional information for the skip connections is tried to be directly acquired by the imitating encoder instead of the contracting path of the U-Net module. Additionally, Ronneberger et al.'s U-Net [RFB15] and Oktay et al.'s ACNN [OF+18] are used as further baseline architectures. Each architecture is trained in a leave-one-out manner, such that any image, that is not the test image can be used for training and validation.

Data

For this series of experiments, 38 fluoroscopic x-ray images of the femur from 38 different patients are used. The x-ray images were recorded during clinical routine for necrosis treatment and are provided by the Department of Orthopaedics and Trauma Surgery at the University Hospital Essen. The images are heterogeneous, as they vary in size and spacing, resulting in very different FOVs. The image size ranges from (2140×1760) to (4248×4200) pixels, and the spacing varies between $(0.1 \times 0.1)mm^2$ and $(1 \times 1)mm^2$.
Furthermore, in all images only the femur side is labeled which is supposed to receive necrosis treatment. Therefore, in some images, the right femur is annotated, whereas in others the left femur is labeled. Fig. 4.23 illustrates the range in field of view and the mix of right and left femur labels. To simplify the segmentation

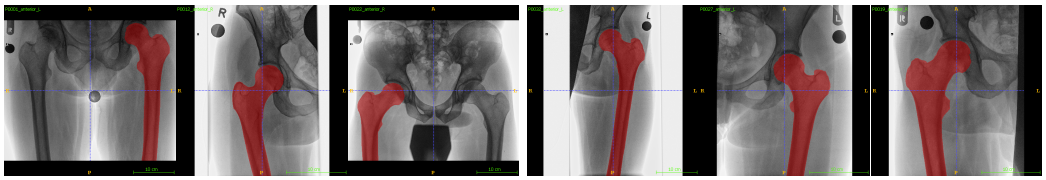


FIGURE 4.23: Variation of labeled fluoroscopic x-ray images.

problem, all images, that have labels for the left femur, are flipped for training, validation and testing, reducing the task to the extraction of the right femur. Otherwise, most neural networks would not be able to determine which femur side to extract, as in some images both femur bones are visible, but only one of them needs to be extracted.

Results

Table 4.11 shows the achieved mean DSCs of the considered architectures. It is noticeable that the achieved scores of IED-Net, IE$_2$D-Net, IRE$_3$D-Net, and U-Net are close to each other, each yielding a wide standard deviation. To establish, whether the differences are significant, the Wilcoxon signed rank test [Wil92] is conducted to every possible architecture combination. The estimated p-values, for which the differences are significant, are depicted in Tab. 4.12. Following the convention that

DSC	U-Net	ACNN	IED	IE_2D	IRE_3D
$\emptyset$	0.913 ± 0.11	0.768 ± 0.25	0.918 ± 0.15	**0.928** ± 0.17	0.923 ± 0.16

TABLE 4.11: Resulting mean DSCs.

a p-value smaller than 0.05 yields a significant difference, one can observe that in four cases, a significant difference could not be measured. Both IED-Net and IRE_3D

p**-value**	U-Net	ACNN	IED	IE_2D	IRE_3D
U-Net	-	6.7015e-06	0.2736	5.4260e-04	0.0756
ACNN	6.7015e-06	-	1.1416e-06	7.8917e-07	5.8521e-07
IED	0.2736	1.1416e-06	-	4.8701e-04	0.1845
IE_2D	5.4260e-04	7.8917e-07	4.8701e-04	-	0.0100
IRE_3D	0.0756	5.8521e-07	0.1845	0.0100	-

TABLE 4.12: p-values of every possible architecture combination for DSC difference.

do not perform significantly better than the baseline U-Net. Furthermore, there is not any significant difference between IED-Net and IRE_3D-Net. However, IE_2D-Net yields significantly better results than all remaining architectures.

sHD	U-Net	ACNN	IED	IE_2D	IRE_3D
$\emptyset$	259.44 ± 345.18	1462.01 ± 1056.64	317.27 ± 402.74	**203.29** ± 330.54	1470.69 ± 963.86

TABLE 4.13: Resulting mean symmetric Hausdorff distances.

Table 4.13 depicts the mean symmetric Hausdorff distances over all test images. Surprisingly ACNN and IRE_3D-Net yield significantly larger mean Hausdorff distances than the remaining architectures (see Table 4.14). Apparently, ACNN has difficulties in handling the different FOVs, which is propagated to IRE_3D-Net, which partly consists of the ACNN.
Although achieving a better mean DSC, IED-Net's mean sHD is larger than U-Net's. As shown in Table 4.14, the difference is however only marginal and not significant. The IE_2D-Net on the other hand is able to significantly outperform the remaining architectures, also regarding sHD.

p-value	U-Net	ACNN	IED	IE_2D	IRE_3D
U-Net	-	4.009e-07	0.7918	0.0013	8.5000e-07
ACNN	4.009e-07	-	7.3256e-07	1.250e-07	0.9218
IED	0.7918	7.3256e-07	-	0.0026	5.4555e-06
IE_2D	0.0013	1.250e-07	0.0026	-	4.3255e-07
IRE_3D	8.5000e-07	0.9218	5.4555e-06	4.3255e-07	-

TABLE 4.14: p-values of every possible architecture combination for sHD difference.

For a qualitative illustration of the considered architectures, Fig. 4.24 shows exemplary segmentation predictions of these architectures. The high Hausdorff distances is clearly reflected in the ACNN predictions and partly in the IRE_3D-Net results.

Moreover, U-Net appears to have difficulties in capturing details of the femur, although most of the area is correctly estimated. Similar to IED-Net, the femur boundaries appear not as smooth as it should be. For the IE_2D-Net architecture, however, the femur contours are a lot smoother for these exemplary predictions. It therefore seems to be able to handle the different FOVs in the data set better than the remaining architectures.

4.4.6 Summary

In this section, the IRE_3D architecture is presented, combining on the IE_2D design and Oktay et al.'s ACNN. Its performance on cross-modality learning tasks is demonstrated in scope of the *CHAOS* challenge, and its applicability in one-shot segmentation settings is evaluated on the example of liver segmentation in CT. Here the network design leads to significant improvements compared to U-Net, however, significant differences to IE_2D-Net can only be observed for the situs inversus case regarding symmetric Hausdorff distance. Ablation studies on the example of 2D femur extraction from fluoroscopic x-ray images indicate, that in a more general segmentation setting without one-shot or cross-modality learning conditions IE_2D-Net should be preferably utilized, as its segmentation predictions are significantly better than IRE_3D-Net's predictions. Furthermore, the U-Net module for hierarchical features in the IE_2D-Net seems to be a crucial component for general segmentation settings with varying FOVs, as it significantly improves the segmentation predictions compared to IED-Net.

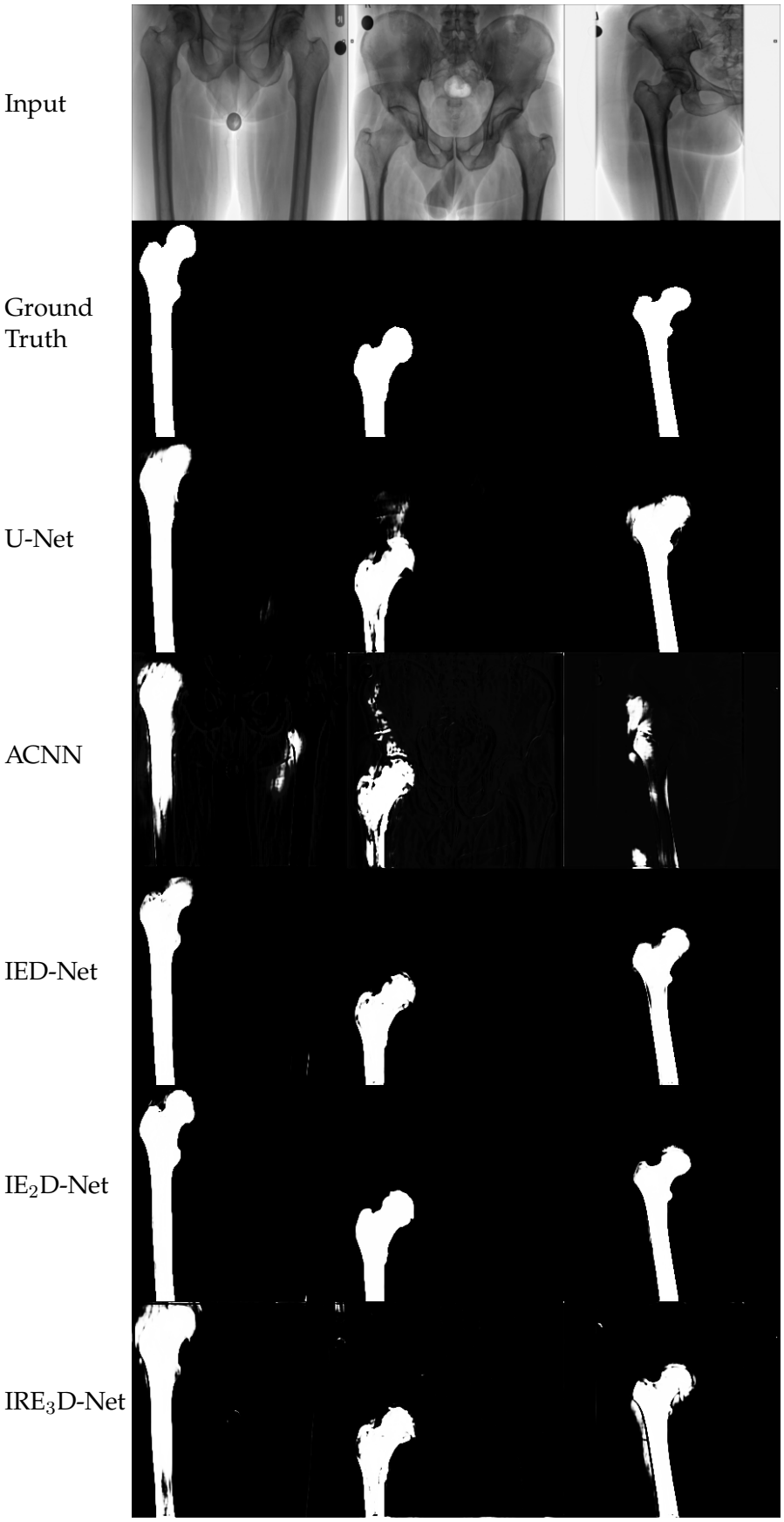


FIGURE 4.24: Qualitative illustration of the results of the considered architectures.

4.5 Shape and Contour Priors for Zero-Shot Domain Adaptation

This section discusses strategies to enforce shape aware learning, specifically in the *domain adaptation* setting. The notion of infusing shape and low level contour information into the training process is investigated. Furthermore, the idea of augmenting the training data by color map transformations is proposed, to abstract from intensity based towards shape aware features. A major part of its content has been previously published in [PDP20].

The concept of domain adaptation is usually necessary when deep learning architectures are trained in a particular domain, the *source domain*, but are applied in a similar but different *target* domain during inference. This scenario is of crucial practical importance, especially in the medical context, as different hospital departments tend to use different imaging modalities and protocols in their clinical routine. Thus, training a model with source data from one department may not be sufficient for application in another institution with different target data. Acquiring new labeled training data to train models for each target domain and retraining existing models with this new data, is however a repetitive and expensive task, that can be avoided using domain adaptation strategies. The difference between source and target domain can be marginal, e.g. when different patients are considered or when different machines are used to acquire the images for the same modality, e.g. CT. This so called *domain shift* can, however, be more drastic, e.g in case of cross-modality domain adaptation cases, in which a network, which is trained on one modality, is supposed to be applied in a different modality. Fig. 4.25 shows exemplary imaging protocols, that are used in clinical routine. It is noticeable, that different MRI protocols yield different intensity distributions and different texture properties of the organs. Another challenge is the field of view property, as illustrated between the first three examples and the last image, where the ratios between patient body area and background can be very different.

The case of *zero-shot domain adaptation* or *domain generalization*, in which the target domain is unknown during training, is of particular interest. The main idea of this section is that shape information may help in bridging the gap of occurring domain shifts from source training to target application domains, independent of the actual target domain. A particular focus of this section is the domain generalization from MR to CT volumes on the example of 3D liver segmentation. Most adaptation strategies make use of target domain samples and often additionally incorporate the corresponding ground truths from the target domain during the training process. In contrast to these approaches, the feasibility of training a deep learning model solely on source domain data is investigated.
To compensate the missing target domain data, prior knowledge about similarities in imaging modalities is used to steer the model towards more general features during the training process. Similarities regarding contour progression and shape information across imaging modalities (see Fig. 4.25) is of particular interest. The presented strategy makes use of fixed Sobel kernels to enhance contour information and it applies anatomical shape priors, learned separately by a convolutional autoencoder. Furthermore, a color map augmentation strategy is proposed to abstract from texture based features towards shape aware features.

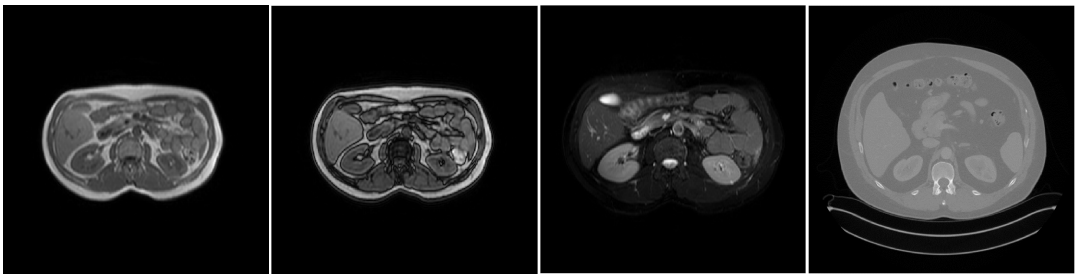


FIGURE 4.25: Comparison of different imaging protocols. Different hospitals prefer different imaging protocols and modalities. From left to right: T1 weighted in-phase MRI, T1 weighted out-of-phase MRI, T2 weighted SPIR MRI, CT image.

4.5.1 Methods

As Ronneberger et al.'s U-Net [RFB15] yields competitive results in various medical image segmentation applications, the proposed method is based on this architecture. The main idea is to make use of shape information both by providing low-level contour information to the base segmentation network and also by regularizing the learning process by means of more abstract and learned shape priors. The proposed IE_2D and IRE_3D architectures from sections 4.3 and 4.4 are however not applicable to this zero-shot domain adaptation setting. The imitating encoder tries to project the input image to a latent representation that is similar to the latent representation of the corresponding ground truth map. Therefore, this encoder is fed with images from the source domain during training and can therefore only handle source domain inputs instead of target domain images.

Base Segmentation Network

Çiçek et al.'s [ÇA+16] 3D extension of Ronneberger et al.'s U-Net [RFB15] is used as the base segmentation architecture, since it is a fully convolutional network (FCN) [LSD15] designed specifically for segmentation tasks in medical image computing, and has achieved impressive results in various segmentation tasks [Ise+21]. Two additional modifications to the base architecture are investigated. On the one hand, an edge enhancement (EE) module is introduced to provide low-level contour information to the segmentation network. On the other hand, following Oktay et al.'s suggestion [OF+18], a convolutional autoencoder is trained beforehand to generate a data driven compact high-level shape representation, in order to add a regularization term to the segmentation loss function. This method has also been utilized in the context of the previous sections regarding IE_2D- and IRE_3D-Net.

Low-Level Contour Information

The aim is to train the segmentation network on a source domain, such that the same trained architecture can be reused on a different target domain without any modifications. Therefore, the network is steered into learning more general features, which are applicable in an inter-domain manner. Considering Geirhos et al.'s findings [Gei+18], one major objective is to make sure that the network does not learn texture based features, as soft tissue texture varies significantly depending on the modality and on the protocol. Fig. 4.25 shows the difference in intensity appearance across modalities and even acquisition protocols within the same modality. Instead,

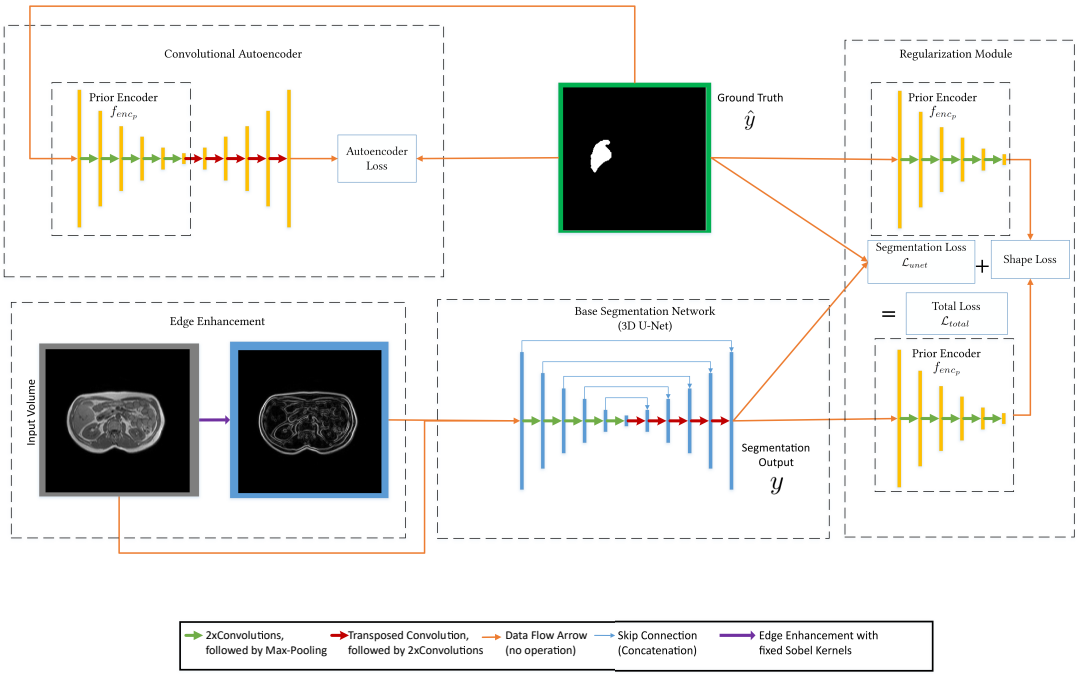


FIGURE 4.26: The network architecture consists of two separate networks. The first autoencoder network learns a latent representation of the ground truth, that includes information about anatomical priors, and is trained separately from (and previous to) the second network. The second network is trained for segmentation and is regularized by a shape loss term, that is computed by means of the trained autoencoder's encoder component. For this regularization, the trained autoencoder's encoder is reused. The segmentation network is extended by an edge enhancement component, comprising fixed Sobel kernels. Adapted from [PDP20], copyright ©2020 IEEE.

the aim is to make use of the observation, that in medical images structures usually show lower variation in perspective, shape and topographic composition than in natural images. Particularly contour progression and shape information should be of special interest. Therefore, an edge enhancement component is introduced to the contracting path, consisting of 3D Sobel kernels in each direction, as can be seen in Fig. 4.26 on the lower left side. Let I denote the input volume, and S_x, S_y, S_z the aforementioned Sobel kernels. Then the edge enhancement is achieved by calculating the gradient magnitude volume

$$|\nabla I| := \sqrt{(I * S_x)^2 + (I * S_y)^2 + (I * S_z)^2}, \tag{4.11}$$

where $*$ is the convolution operator. The resulting gradient magnitude volume is concatenated to the original input volume and passed on to the segmentation network, as can be seen in Fig.4.26 at the bottom. The idea is that this additional low level contour information may steer the network towards more texture independent, but more shape relevant features. Because of the different modality and protocol specific appearance of anatomical organs, the resulting gradient magnitude will also differ between modality and protocol. However, the information about organ contour progression is similar between all domains and highlighted in the gradient magnitude volume, as illustrated in Fig. 4.27, where exemplary gradient magnitude slices from different protocols and modalities are shown. These images are depicted

on a logarithmic scale to make available contour information more visible.

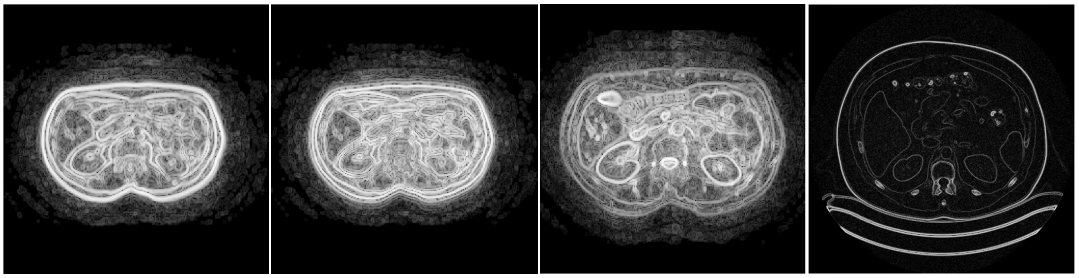


FIGURE 4.27: Although gradient magnitudes may differ between protocols and modalities, a similar contour progression is highlighted at the organ boundaries for the gradient magnitude volumes of the exemplary T1 weighted in-phase MRI, T1 weighted out-of-phase MRI, T2 weighted SPIR MRI, CT image (from left to right).

High-Level Shape Representation

As a second modification, an additional convolutional autoencoder (CAE) is trained, following Oktay et al.'s [OF+18] suggestion to incorporate more abstract shape priors into the segmentation network. This CAE is trained by optimizing its loss function $\mathscr{L}_{auto}$. The CAE consists of a prior encoder f_{enc_p} and a decoder component. A sufficiently trained encoder f_{enc_p} is capable of compressing relevant anatomical information, implicitly encoded in the ground truth, into a compact representation in latent space. The decoder component on the other hand is able to reconstruct the ground truth from this compressed representation. Crucial anatomical information about shape and topographical relationships is embedded in this high-level representation. This compressing property is leveraged to compare the embedded information in the segmentation output with the ground truth's compressed representation. Thus, f_{enc_p} is reused to project both the ground truth and the segmentation output into latent space, as shown in Fig. 4.26 on the right. The L_1 norm is used to measure the difference in latent representation and add this term to the segmentation loss $\mathscr{L}_{unet}$ for shape regularization, weighted by a factor $\lambda_{reg} > 0$, i.e.:

$$\mathscr{L}_{total}(y, GT) := \mathscr{L}_{unet}(y, GT) + \lambda_{reg}||f_{enc_p}(y) - f_{enc_p}(GT)||_1. \tag{4.12}$$

All in all, the proposed method consists of 2 training stages. In the first stage the CAE is trained by minimizing $\mathscr{L}_{auto}$, and in the second stage the segmentation network is trained by optimizing $\mathscr{L}_{total}$. In the second stage, only weights from the segmentation network are adapted, whereas the weights from f_{enc_p} and the Sobel kernels are kept fixed.
In contrast to related work regarding domain adaptation (see section 4.1), no prior target domain information is used during training. Instead, this method solely relies on the data material of the source domain. A thorough search of the relevant literature yielded that this is the first attempt at using contour and shape information in the context of zero-shot domain adaptation for medical segmentation tasks.

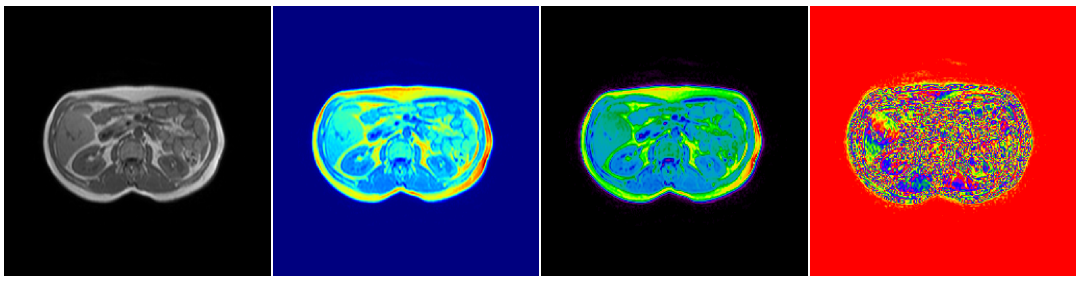
Enforcing Shape Awareness through Color Augmentation

Besides directly infusing shape aware Sobel features, a more implicit method lies in the enforcement of shape aware feature learning by means of appearance augmentations of the training data. Geirhos et al. [Gei+18] show that by means of style

augmentation, classification CNNs abstract from intensity and texture based features towards shape aware features. Style augmentation is the process of injecting random styles into an input image, such that the image style appearance is deviated, whereas the overall composition and content stays the same. In the context of medical image segmentation, Hesse et al. [Hes+20] use the same idea of applying style (and intensity) augmentation to bridge the domain shift between data sets of the same modality, but with different sequences. In scope of this thesis, a novel color augmentation procedure is proposed, in which existing *color maps* are used to augment the appearance of the source domain input volumes. A color map is a mapping

$$CMAP : \mathcal{I}_{gray} \rightarrow \mathcal{I}_{color}$$

from the space of gray scale images $\mathcal{I}_{gray}$ to the space of color images $\mathcal{I}_{color}$. Each gray scale value is mapped to a specific color, depending on the chosen color map. Therefore, a one dimensional gray value is mapped to a three dimensional RGB vector. *Color augmentation (CA)* is realized by randomly applying a color map on the intensity image. One major advantage of using color augmentations instead of style augmentations is the adhoc applicability on 3D volumes. Moreover, color augmentations can be achieved in constant time for each pixel, since the intensity value is directly mapped to a color by the color map. A further benefit lies in the reduction of required GPU memory compared to style augmentation, which makes use of pretrained CNNs for style infusion.
Similar to the concept of *filter banks*, in which selected filters are designed for feature extraction in images, only 31 selected color maps are used for color augmentation, which are taken from all available color maps, provided by *matplotlib's* Python package [Hun07]. The selection has been applied manually based on visual suitability.



(a) original image (b) jet (c) nipy_spectral (d) prism

FIGURE 4.28: Suitability of color maps. Some color maps are more suited (b) and (c) than other (d) for color augmentation of the original input image (a).

Figure 4.28 demonstrates that a selection of available color maps is a necessary step, as some color maps seem to be more suitable for augmentation than others. All selected color maps and their color range are depicted in Fig. 4.29.

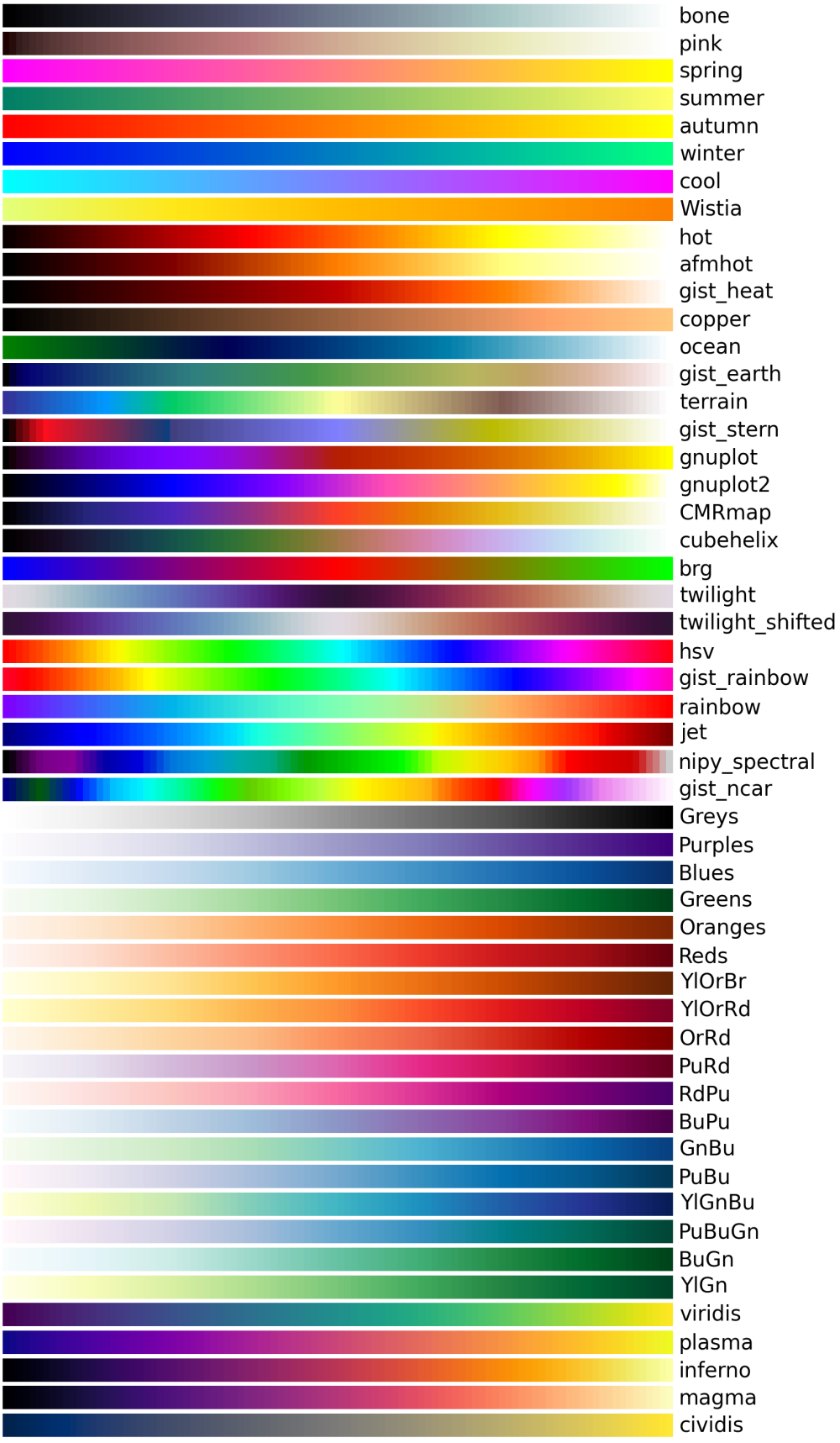


FIGURE 4.29: Selected color maps for color augmentation.

4.5.2 Experiments

The proposed modifications are investigated on the example of 3D liver segmentation. In the scope of this thesis, the source domain consists of T1 and T2 weighted MR volumes, whereas the target domain comprises CT scans.
To investigate the impact of the changes to the base architecture, ablation studies are conducted on the effects of

- adding the low-level edge enhancement (EE) for the extraction of contour information
- adding the more abstract shape priors (SP) to the base segmentation network
- random color augmentation during training

As an upper baseline, the segmentation results of the proposed method are compared to a fine-tuning strategy, in which the base segmentation network is pretrained with MR images and fine-tuned with a fraction of the target domain CT data.
An U-Net trained on synthesized CT volumes, which are generated from MR volumes, serves as a second baseline, since this strategy is commonly used for domain adaptation settings (see section 4.1.3). These *pseudo CT (pCT)* volumes are created using Zhu, Isola et al.'s [Zhu+17; Iso+17] cycleGAN implementation in PyTorch.
These last two baselines are, however, not zero-shot domain adaptation methods, as they incorporate prior target domain knowledge into their training process. As evaluation metric for the segmentation predictions, the Dice Similarity Coefficient (DSC) is used for all experiments.

Data

For training and validation, the labeled MR data sets from the Combined Healthy Abdominal Organ Segmentation (CHAOS) challenge [SK+19; Kav+21] (see section 4.4.3) are used. They comprise two different MRI sequences, i.e T1 weighted in-phase and out-of-phase volumes and also T2 weighted SPIR data sets, resulting in 60 labeled data sets in total. The MR data sets have been acquired by a 1.5 Tesla Philips MR scanner. Each volume consists of 26 to 50 axial slices with a slice size of 256×256. The trained models are tested on the 20 provided CT scans of the same challenge. Furthermore, 43 CT volumes from the Cancer Imaging Archive [CV+13; RF+16; RL+15] (TCIA) and 47 CT volumes of the Beyond the Cranial Vault (BTCV) segmentation challenge [LX+15; XL+16] (see section 4.4.4) are used as additional CT target domain data to test the architectures. The supplementary ground truth segmentation for both data sets, provided by Gibson et al. [GG+18], are used to evaluate the network predictions.
For the fine-tuning baseline strategy, 10 CT volumes from the CHAOS challenge are used during training, which are excluded from the testing set. This needs to be considered when comparing the achieved DSC results for the CHAOS data sets.

Implementation Details

To fit the 3D architecture into memory, the MR and CT volumes are resized to an input size of $128 \times 128 \times 96$. The network is implemented in Tensorflow 1.12. In all experiments, the training volumes are augmented by means of random translation and rotation. Starting with 32 kernels for each convolutional layer in the first scale level of each encoder/contracting path, the number of kernels for each scale level on the encoding/contracting side is doubled, whereas the number of kernels on the

expansive/decoding side for both segmentation network and autoencoder is halved. A kernel size of $3 \times 3 \times 3$ for every convolutional layer and $2 \times 2 \times 2$ max-pooling is chosen. The optimization is performed with an Adam Optimizer with an initial learning rate of 0.001. With a batch size of 1, the networks are trained for 200 epochs. The model with best validation loss is chosen for evaluation on the test sets. The weight of the regularization term is set to $\lambda_{reg} = 0.001$, as suggested by Oktay et al. [OF+18]. Since

$$||f_{enc_p}(y) - f_{enc_p}(GT)||_1$$

and $\mathscr{L}_{unet}$ from Eq. (4.12) may not be in the same range, it needs to be noted, that λ_{reg} is not only a weighting factor, but also includes a normalization component.
The training is performed subsequently, i.e. the autoencoder is trained first for 200 epochs, followed by the segmentation network, also for 200 epochs.
In case of the fine-tuning baseline implementation, first the U-net is trained for 200 epochs on the source domain. The model with least validation loss is then fine-tuned for an additional 200 epochs on the CT target domain. If used, the color augmentations are applied on 50% of the training volumes on the fly.
For the GAN based baseline, pseudo CT scans are generated from the available source MR volumes by means of Zhu, Isola et al.'s [Zhu+17; Iso+17] cycleGAN implementation in PyTorch. Since the original cycleGAN was designed for 2D images, the 2D slices of the MR volumes are used to generate pseudo CT slices. These are stacked to 3D volumes and used to train a 3D U-Net implementation for 200 epochs. All experiments are conducted on a NVIDIA GTX 1080 TI GPU.

Ablation Studies

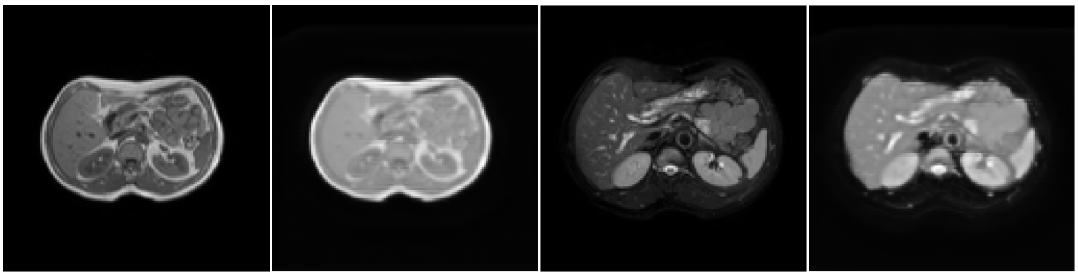
For the ablation studies, the U-net implementation is equipped with each proposed modification, i.e. shape priors (SP), edge enhancement (EE), and color augmentation (CA), individually and in various combinations. Table 4.15 shows the achieved DSCs for each experiment and each target domain. As expected, the stand-alone U-Net performs worst, as the MR source distribution is significantly different from the CT target distribution. Surprisingly, the fine-tuning strategy is outperformed by the combination of edge enhancement and color augmentation in two of three target domains, although CT volumes are incorporated into the fine-tuning process during training. It is noteworthy, that adding low level contour information by means of simple edge enhancement (EE) significantly increases the DSC for all three target domains and improves the segmentation results reasonably more than adding abstract shape constraints in terms of shape regularization, at least for the 3D case. A similar improvement can be observed for the sole integration of color augmentation.
The best overall results are achieved by combining edge enhancement with shape constraints and color augmentation, outperforming all other combinations and baselines with a mean DSC of 0.798 ± 0.14.
Unexpectedly, the U-Net trained with pseudo CTs by Zhu, Isola et al.'s [Zhu+17; Iso+17] cycleGAN implementation yields worst segmentation results in all target domains. This may be due to the fact, that the generated pseudo CTs might be of inferior quality, as the source data needed to be converted to *.png* format and resized to a low input size of $128 \times 128 \times 96$, which is more prone to blurring effects. Additionally, the pseudo CT slices do not seem to be able to represent the target domain sufficiently, also revealing artificial artifacts, as shown in Fig. 4.30. Here, it can be observed that the generated pseudo CTs are representative in the overall intensity

	CHAOS	TCIA	BTCV	$\emptyset$
U-Net	0.496 ±0.17	0.582 ± 0.12	0.301 ± 0.18	0.446 ± 0.20
U-Net+SP	0.594 ±0.16	0.604 ± 0.10	0.335 ± 0.23	0.487 ± 0.22
U-Net+EE	0.720 ±0.19	0.762 ± 0.08	0.519±0.13	0.650 ± 0.17
U-Net+CA	0.754 ±0.17	0.764 ± 0.12	0.397±0.29	0.605 ± 0.28
U-Net+EE+SP	0.753 ±0.13	0.817 ±0.13	0.551±0.29	0.692 ±0.25
U-Net+EE+CA	**0.820** ±0.16	0.861 ±0.06	0.669±0.13	0.772 ± 0.14
U-Net+EE+CA+SP	0.814 ±0.19	**0.862** ±0.07	**0.730**±0.12	**0.798** ±0.14
U-Net+CT Fine-tuning	*0.916* ±0.06*	*0.828* ±0.07	*0.662* ±0.15	*0.766* ±0.15*
U-Net on pseudo CT	0.359 ±0.18	0.527±0.17	0.273 ±0.19	0.391 ±0.21

TABLE 4.15: Achieved mean DSCs in each CT test data set and the average DSC over all test sets. The upper table shows the results from the ablation study with shape priors (SP), edge enhancement (EE), and color augmentation (CA), whereas the bottom part shows the achieved DSCs from the baseline implementations. *10 CT data sets from the test set are used for fine-tuning, thus, only the remaining 10 CHAOS data sets are considered for evaluation. Based on [PDP20], copyright ©2020 IEEE.

appearance, however, structure boundaries are either blurry (Fig. 4.30 (b)) or even incorrect (Fig. 4.30 (d)). This may be due to the fact that the abdominal image-to-image translation task seems to be very different to e.g. cranial pseudo CT generation because of the larger variation in present organs and their appearance in CT slices.

Figure 4.31 depicts exemplary segmentation results from the evaluated test cases. The first three columns show one example from each data set, whereas for the last column an example is chosen, in which the best combination does not perform well. In this figure, it is clearly visible that incorporating low level contour information and color augmentation lead to a considerable segmentation improvement. It is particularly noteworthy, that the addition of edge enhancement yields a more precise contour progression, especially in the area of the portal vein (first and second columns), where the incision of the liver is correctly followed. In combination with Oktay et al.'s [OF+18] anatomical shape constraints this yields a promising strategy for domain adaptation in medical image computing without even having to consider the target domain in the training process.



(a) T1 MRI (b) pCT (c) T2 MRI (d) pCT

FIGURE 4.30: Exemplary pseudo CTs. (a) and (c) show the original images and (b) and (d) the corresponding pseudo CTs, generated by cycleGAN [Zhu+17; Iso+17].

4.5.3 Summary

In this section, three concepts to consider shape information for domain generalization are presented. The first concept uses edge information to directly infuse low level shape aware features into the segmentation network. As a second idea, Oktay et al.'s ACNN [OF+18] is applied for shape regularization. The final suggestion consists of using color maps for fast color augmentations to indirectly enforce the abstraction of texture and intensity based to shape aware features.
In ablation experiments, it is demonstrated that a combination of all proposed strategies even outperforms the upper baseline fine-tuning approach. This is especially noteworthy, as no target domain data is considered in any way during training.
Surprisingly, the adversarial cycleGAN baseline results in the worst DSCs in the comparison.

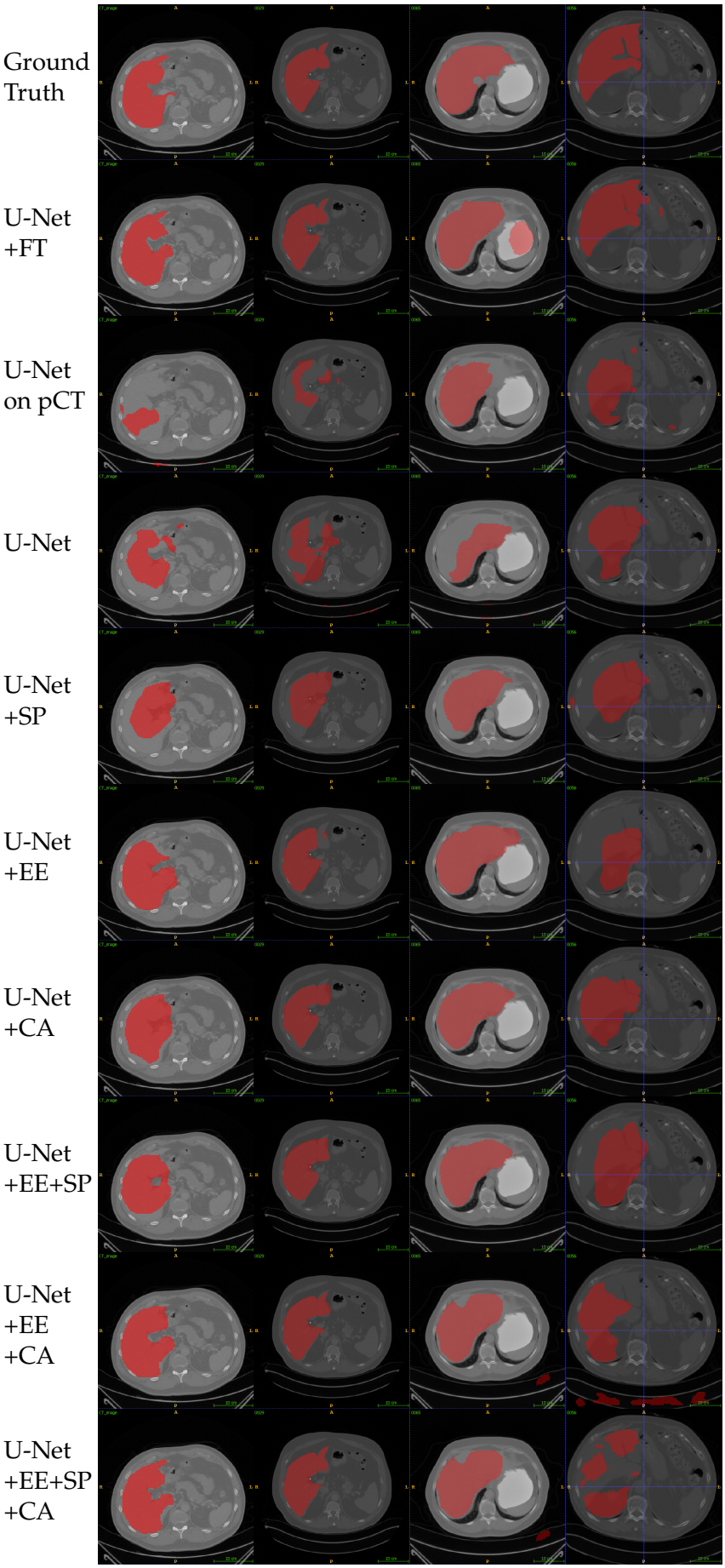


FIGURE 4.31: Exemplary slices from the three target domains CHAOS, TCIA, BCTV and one exemplary difficult slice (from left to right column).

4.6 Conclusion

This chapter presents multiple strategies to incorporate prior shape information into deep learning architectures.

- The cascaded convolutional distance transform is an adhoc method to realize distance transforms in deep learning architectures. The conducted studies show significant improvements compared to the reference U-Net on the example of the segmentation of thoracic organs, particularly regarding Hausdorff distance.
- A further approach lies in the imitation of the compression operation in a convolutional autoencoder. This general idea yields two architectures, namely the IE_2D-Net and IRE_3D-Net.
  - The IE_2D-Net shows promising but marginal improvements to U-Net in a slice-wise approach for volumetric image segmentation.
  - In the ablation studies for IRE_3D-Net, IE_2D-Net shows clearer improvements for 2D segmentation settings, in which shape appearance shows consistency, but the structure of interest's (e.g. the femur's) FOV may strongly vary.
  - Therefore, it is recommended to use IE_2D-Net for segmentation tasks, in which the shape of the structure of interest generally stays consistent across input samples, such as direct 3D segmentation or 2D segmentation tasks, where a decomposition of the input to slices or patches is not required.
  - For a slice-wise volumetric approach, e.g. for high resolution extractions with limited GPU memory, a well configured U-Net approach suffices, especially because of its more straight forward training procedure.
  - Furthermore, IRE_3D-Net shows superior performance in cross-modality learning in an unrestricted competition environment. Moreover, it shows significant improvements to U-Net in one-shot segmentation settings, especially in cases where training and test sample show only little overlap.
- Finally, three shape infusing strategies for domain generalization are proposed and evaluated, all showing superior results to a baseline U-Net in ablation studies. A combination of these strategies is even able to outperform the upper fine-tuning baseline.

Chapter 5

Topographical Priors in Deep Learning Architectures

Chapters 3 and 4 pay particular attention to the concept of *shape* as an instance of anatomical priors, since the shape of many structures of interest within the human body show only little variation across patients and imaging modalities. There are, however, also structures of irregular shape, for which the presented methods of previous sections do not apply. For these kind of structures, another form of anatomical prior may be beneficial for a better extraction performance. This chapter particularly focuses on the usage of *topographical* information of the structures of interest in deep learning architectures. In contrast to shape priors, the key information of topographical priors lies in the prior knowledge about the position of its occurrence within the human body instead of its shape and contour. In this chapter, section 5.1 demonstrates the incorporation of topographical priors into a deep learning architecture on the clinical example of the extraction of avascular necrosis of the femoral head. Then, section 5.2 makes an excursion to a semi-supervised caries detection approach, in which the concept of topographical priors aids in a better detection performance. The content of this section is based on the following publications, for which content reuse is permitted:

> **[Pha+20]** Duc Duy Pham et al. "Multitask-Learning for the Extraction of Avascular Necrosis of the Femoral Head in MRI". In: *Bildverarbeitung für die Medizin 2020*. Springer, 2020, pp. 150–155
>
> **[Pha+21]** Duc Duy Pham et al. "Fully vs. Weakly Supervised Caries Localization in Smartphone Images with CNNs". In: *Pattern Recognition. ICPR International Workshops and Challenges*. Springer International Publishing, 2021, pp. 321–336

5.1 Extraction of Avascular Necrosis of the Femoral Head in MR Images by Multitask Learning

In this section, a 2D deep multitask learning approach is presented, which addresses the task of extracting small structures with irregular shape, leveraging prior knowledge about the probable place of occurrence. This is done on the example of avascular necrosis of the femoral head (AVNFH) in MR volumes.

5.1.1 Motivation

Necrosis is a disease process of non-programmed premature cell death. In case of the hip joint, various causes can lead to avascular necrosis of the femoral head (AVNFH), such as a disruption of the blood supply by means of traumatic injuries, physical obstructions or metabolic issues. A collapse of the femoral head can lead to functional damage of the hip joint. A precise assessment of the necrotic area helps in operative planning and may prevent unnecessary total endo-prosthesis (TEP). Since manual segmentation is expensive and time consuming, there is a high demand for computerized fully automated methods. AVNFH presents large variability in its shape and appearance, which makes the segmentation a challenging problem. Therefore it cannot be addressed by the previously presented methods of this thesis.

5.1.2 Related Work

Zoroofi et al. [Zor+01] present a semi-automated necrosis segmentation pipeline of traditional image processing methods, using histogram based thresholding in a region of interest (ROI) and ellipse fitting in oblique slices, which are set to be perpendicular to the femoral collum. Similar approaches, that also estimate the ROI beforehand, can be observed for other segmentation tasks, which deal with the extraction on small structures. For the segmentation of brain lesions Song et al. [Son+16] use a two-stage strategy in first approximating the ROI by means of thresholding and GrowCut, in order to use a random forest classifier afterwards for pixelwise classification.

In the scope of deep learning, the usage of these kind of multi-stage approaches can also be observed for the extraction of small and irregular structures. For the extraction of liver lesions both Christ et al. [Chr+16] and Vorontsov et al.[Vor+18] essentially segment the liver first, in order to find the lesions afterwards. Hatamizadeh et al. [Hat+19] combine the output of a Convolutional Neural Network with an extended Level Set method for the refinement of the initial lesion segmentation.

In this section, it is investigated, how well a variant of Ronneberger et al.'s U-Net [RFB15] deals in segmenting rather small structures in large MR images. Additionally, an alternative deep 3-branch multitask fully convolutional architecture is proposed, which combines the tasks of image reconstruction, necrosis extraction, and putting the necrosis into topographical context. Here the task of necrosis extraction is the main objective, which is aided by the remaining two auxiliary tasks.

5.1.3 Methods

Deep 3-Branch Multitask Fully Convolutional Architecture

In multitask learning, several tasks are addressed simultaneously in order to facilitate common properties across related tasks. In this particular deep learning setting, a joint convolutional encoder is proposed, followed by three task-specific convolutional decoder branches. The main objective is represented by a *segmentor* branch that aims at segmenting the femoral necrotic area.

Let y_{seg} denote the output of the segmentor branch. Then, an auxiliary task of reconstructing the input from a latent representation by means of an *autoencoder* branch can be defined. The objective of the autoencoder is, however, restricted to a partial image reconstruction in the neighborhood of the necrotic area. The idea is to particularly enforce a compact latent representation, from which the autoencoder branch can mainly reconstruct the necrotic area. The result of this branch is depicted as y_{auto}. Furthermore, an additional task of approximating the location of the topographical neighborhood of the femoral necrosis is introduced. The intention is to learn features that are activated by inter-patient consistent anatomical structures within that location. In this scenario, the femoral head would be such anatomical structure, that usually constrains the topographical location of AVNFH and that can be consistently found across patients.
The output of this *topographical* branch is denoted as y_{topo}, for which the pixel values range between 0 and 1.

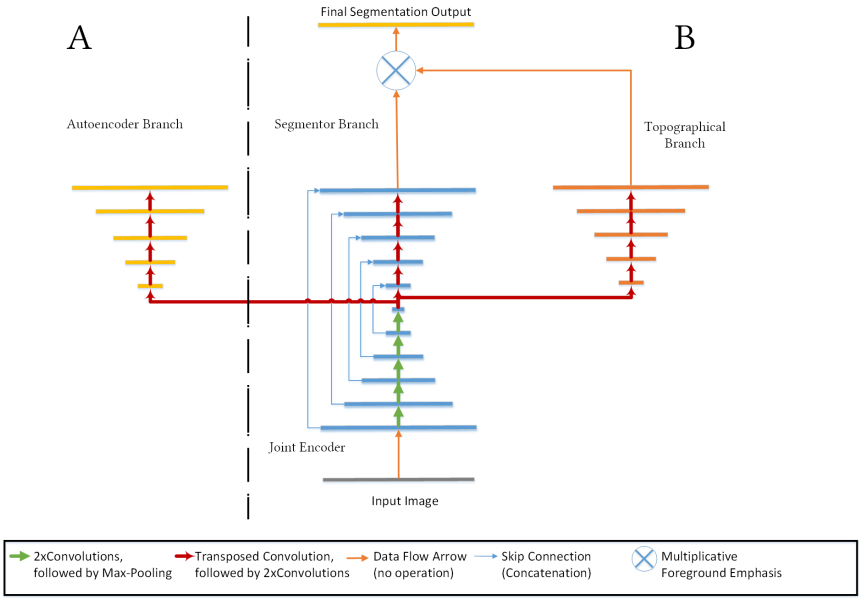


FIGURE 5.1: Multitask architecture consisting of joint encoder and 3 different decoder branches. Main objective is the segmentation, auxiliary tasks are image reconstruction and topographical neighborhood localization. For the final segmentation the segmentor branch is fused with the topographical branch (side B). The autoencoder output is omitted for inference (side A). Modified from [Pha+20].

As shown in Fig. 5.1, skip connections between the joint encoder and the segmentor

branch are used to improve the fine pixel localization capabilities of this decoder, basically rendering this cascade a U-Net variant. Skip connections to the topographical branch are not used, since the rough topographical approximation of the surroundings does not need any fine-resolution information. The autoencoder branch does not receive any skip connections, either, as these would allow the decoder to copy the relevant information for the image reconstruction task from the shallow encoder layers. In both cases the necessity to compress the input image into a compact representation in latent space is therefore enforced.

For the final segmentation, the foreground output of the topographical branch is multiplied with the segmentor branch output to emphasize on this topographical neighborhood, i.e.

$$y_{final}^{(fg)} := y_{topo}^{(fg)} \otimes y_{seg}^{(fg)}, \tag{5.1}$$

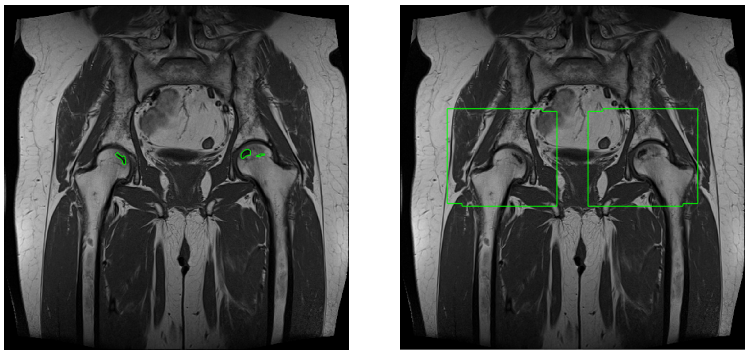
where $\otimes$ denotes the Hadamard product and $y_{(\cdot)}^{(fg)}$ the foreground necrosis channels of the output $y_{(\cdot)}$. The background channel of the final output is adjusted to

$$y_{final}^{(bg)} := 1 - y_{final}^{(fg)},$$

such that $y_{final}^{(fg)}, y_{final}^{(bg)} \in [0, 1]$. In the remainder of this section, leaving out any superscript annotation denotes using all channels.

Training

To train the topographical branch, a ground truth for the desired neighborhood is required. For this, the original ground truth necrosis segmentation GT_{seg} is relaxed to squared environments around the necrotic area (Fig. 5.2). Since a necrosis-independent localization is desired, each box is mirrored along the vertical axis, as can be seen in Fig. 5.2 (b). The overlay of ground truths from both sides is particularly visible in the stepped upper and lower bounding lines. This relaxed ground truth GT_{topo} is used to train the topographical branch by means of the dice loss (see Eq.(2.2) in chapter 2.4.1). Using this relaxed ground truth, the topographical branch learns to locate the neighborhood of the femur head. It is, however, not able to distinguish between necrotic and non-necrotic femur heads, as the this information is not provided by the relaxed ground truth information.



(a) (b)

FIGURE 5.2: Construction of topographical ground truth: (a) Necrosis ground truth, (b) relaxed topographical ground truth.

Since the necrotic area only consists of a small fraction of the whole image, only the foreground pixels of the segmentation output for the dice loss of the segmentor branch are considered, i.e.:

$$\mathscr{L}_{\text{seg}} := 1 - \frac{2 \cdot \sum_p GT_{seg}^{fg}(p) \cdot y_{seg}^{fg}(p) + \epsilon}{\sum_p GT_{seg}^{fg}(p) + \sum_p y_{seg}^{fg}(p) + \epsilon}, \tag{5.2}$$

where $\epsilon > 0$ is a small number to avoid zero in the denominator and p depicts a point in the output/ground truth image.

For the autoencoder branch, the mean squared error over the topographical region of interest is calculated, as an enforcement of a strong reconstruction capability is desired, particularly in the probable area of necrosis, i.e.:

$$\mathscr{L}_{\text{auto}} := \frac{\sum_p GT_{topo}^{fg}(p) \cdot (I(p) - y_{auto}(p))^2}{\sum_p GT_{topo}^{fg}(p) + \epsilon}. \tag{5.3}$$

The autoencoder and segmentor branches' loss functions are combined to a reconstruction-dependent loss function

$$\mathscr{L}_{\text{seg,auto}} := \alpha \mathscr{L}_{\text{auto}} + (1 - \alpha)\mathscr{L}_{\text{seg}}, \tag{5.4}$$

where $\alpha \in [0, 1]$ is a dynamic weighting factor, set to $\alpha := \mathscr{L}_{\text{auto}}$, similar to the proposition in a previous publication [Pha+19b], where a refinement network loss is dynamically weighted by the segmentation performance of a preceding U-Net component. To guarantee $\alpha \in [0, 1]$, the input images are normalized to a range between zero and one. It should be noted, that when taking into account the gradient of $\mathscr{L}_{\text{seg,auto}}$ during training, α is considered a constant, although it is actually dependent on the autoencoder branch. The dynamic weighting scheme encourages the joint encoder to pre-generate features and a latent representation for image reconstruction, before focusing on the segmentation task. Only when $\mathscr{L}_{\text{auto}}$ is small enough, the focus shifts from reconstruction towards segmentation.
Similar to the procedure of the IE$_2$D-Net in chapter 4.3 and the IRE$_3$D-Net in chapter 4.4, the topographical branch is trained separately by a different optimizer. The optimizers are used in an alternating fashion, starting with the topographical branch. It should be noted that the autoencoder branch is only needed for training, being neglected during inference, as indicated in Fig.5.1 by the separation line between side A and B.

5.1.4 Experiments

In the following experiments the proposed architecture is compared to a U-Net baseline, that is trained in a similar and therefore comparable fashion. Dice Similarity Coefficient (DSC) and symmetric Hausdorff distance (sHD) serve as evaluation metrics for the segmentation of the necrotic tissue. Because of the small size of the structure of interest, precision and recall are also depicted for further insights into the segmentation performance.

Data

The evaluation is conducted on twelve coronal T1-weighted MR volumes, each comprising 19 to 30 slices, which are denoted as P1, ..., P12. The MR volumes were

acquired by a *Siemens Magnetom Aera 1,5 Tesla* MR tomograph during clinical routine and are provided by the Department of Orthopaedics and Trauma Surgery at the University Hospital Essen. In contrast to the MR data in chapter 3.3.5, this data set consists of coronal slices and shows a lot more variability in its resolution. Table 5.1 shows the voxel spacing and the volume size of the available patient volumes. It is noteworthy that the original volume sizes are much larger than in chapter 3.3.5, while the depth resolution is much coarser than before.

Patient Volume	Voxel Spacing	Volume Size
P1	$(0.84 \times 0.84 \times 4.8)$	$(512 \times 512 \times 19)$
P2	$(0.85 \times 0.85 \times 6.6)$	$(448 \times 448 \times 20)$
P3	$(1.12 \times 1.12 \times 5.2)$	$(350 \times 448 \times 20)$
P4	$(0.78 \times 0.78 \times 4.4)$	$(512 \times 512 \times 30)$
P5	$(1.19 \times 1.19 \times 5.2)$	$(320 \times 320 \times 25)$
P6	$(0.67 \times 0.67 \times 5.9)$	$(560 \times 560 \times 17)$
P7	$(0.73 \times 0.73 \times 4.6)$	$(512 \times 512 \times 23)$
P8	$(0.43 \times 0.43 \times 4.4)$	$(512 \times 512 \times 20)$
P9	$(0.82 \times 0.82 \times 4.8)$	$(512 \times 512 \times 21)$
P10	$(0.74 \times 0.74 \times 4.2)$	$(512 \times 432 \times 25)$
P11	$(0.79 \times 0.79 \times 5.5)$	$(480 \times 480 \times 30)$
P12	$(1.04 \times 1.04 \times 6.3)$	$(384 \times 384 \times 30)$

TABLE 5.1: Resolution of AVNFH data set.

Implementation Details

As before, a variant of Ronneberger et al.'s 2D U-Net is implemented like in chapter 4.3.2. The presented three branch multitask fully convolutional network (MTL-Net) is implemented accordingly, following the design depicted in Fig. 5.1. For training and inference the MR volumes are processed slice-wise. The coronal slices are resized to an input size of 512×512 and normalized to intensity values between 0 and 1. In a leave-one-out cross validation manner, one patient volume is kept for testing while the networks are trained and validated on the remaining patient volumes. Flipping, rotation and translation are used for data augmentation. For all minimization purposes an Adam optimizer with initial learning rate of 10^{-3} is used, respectively. Tensorflow 1.12 serves as deep learning framework and the experiments are conducted on a NVIDIA GTX 1080ti GPU.

Results

For the evaluation, the 2D predictions are stacked to 3D segmentation maps and compared to the desired 3D ground truths. The achieved mean DSC, precision and recall values for both architectures are depicted in table 5.2 for AVNFH of the right and left femur.

The baseline U-Net achieves a mean DSC of 0.341 on the left and 0.325 on the right femur. The proposed MTL-Net on the other hand results in higher mean DSCs of 0.389 and 0.371, respectively. This is reflected in the average precision values, i.e. 0.214 and 0.474 by the U-Net compared to 0.214 and 0.568, achieved by MTL-Net. Regarding mean recall, the U-Net implementation yields slightly higher values than the proposed MTL-Net. Considering the higher mean precision of MTL-Net, this is

	Left femur		Right femur	
	U-Net	MTL-Net	U-Net	MTL-Net
DSC	0.341 ± 0.396	$\mathbf{0.389} \pm 0.442$	0.325 ± 0.253	$\mathbf{0.371} \pm 0.268$
Precision	$\mathbf{0.214} \pm 0.309$	0.214 ± 0.330	0.474 ± 0.337	$\mathbf{0.568} \pm 0.375$
Recall	$\mathbf{0.151} \pm 0.234$	0.119 ± 0.230	$\mathbf{0.347} \pm 0.282$	0.335 ± 0.237

TABLE 5.2: Mean DSC, precision, and recall from a standard U-Net compared to the proposed architecture. Adapted from [Pha+20].

an indicator for fewer pixel-wise false positive predictions by MTL-Net. Therefore, U-Net appears to be more prone to segmentations with more false positives.
For both architectures, it is striking that the achieved DSC values are very small. This is due to the fact, that necrotic tissue is often very small and does not always occur on both sides. If a network oversees a complete necrotic area or predicts necrosis in an image slice, where is none, the DSC value is drastically reduced to zero. Furthermore, overseeing fractions of the necrotic area has a larger impact on the DSC than it would have on larger organs. Overseeing fractions is, however, more probable since the necrotic areas are very small compared to the whole volume.

Patient Volume	DSC		sHD		AVNFH?
	U-Net	MTL-Net	U-Net	MTL-Net	
P1	0.714	**0.740**	213.97	**15.81**	✓
P2	**0.350**	0.332	189.28	**18.47**	✓
P3	**0.088**	0.040	65.80	**12.21**	✓
P4	0.341	**0.559**	18.03	**14.35**	✓
P5	0	0	**332.96**	Inf	✓
P6	0	0	Inf	Inf	✗
P7	1	1	0	0	✗
P8	0	**1**	Inf	**0**	✗
P9	**0.603**	0	**208.90**	Inf	✓
P10	1	1	0	0	✗
P11	0	0	Inf	Inf	✗
P12	0	0	Inf	Inf	✗
$\emptyset^*$	0.299 ± 0.20	0.279 ± 0.19			

TABLE 5.3: Achieved DSCs and sHDs on left femur for each test patient. $\emptyset^*$ depicts the mean over all patients, that actually have avasculare necrosis on this side.

Table 5.3 shows the achieved DSCs and sHDs of both architectures on the left femur for all test patient volumes in the leave-one-out evaluation. The last column depicts, whether there is any AVNFH on this femur side. For the left side, it is directly observable, that half of the patients do not have any AVNFH on their left femur. These patients are particularly susceptible to very weak DSC scores, which could effect the mean DSC drastically. One example is patient P8, for which MTL-Net correctly predicts zero necrosis pixels, resulting in a very high DSC value of 1, whereas U-Net yields a DSC of zero, since at least one pixel is predicted to be necrotic. On the other hand, patient P9 is an example, for which MTL-Net misses out on a high DSC, since it overlooks the necrotic area. Only considering the patients with AVNFH on their left femur, U-Net's mean DSC would then be 0.299 compared to MTL-Net's mean

DSC of 0.279, which is depicted in Table 5.3 as $\emptyset^*$. Taking a look at the achieved sHDs, however, showcases drastic improvements of MTL-Net compared to U-Net for most patients. It is noticeable that entries show a sHD of infinity (Inf). This is either because the prediction is empty (no necrotic area found at all), although there is necrotic tissue, or vice versa. This would yield the Hausdorff distance of empty to non-empty sets, which is defined a infinity (see chapter 2.4.1). Therefore, mean sHDs cannot be considered in this evaluation.

Patient Volume	DSC		sHD		AVNFH?
	U-Net	MTL-Net	U-Net	MTL-Net	
P1	0.732	**0.738**	202.78	**104.81**	✓
P2	0.575	**0.697**	370.95	**99.58**	✓
P3	0	0	Inf	Inf	✗
P4	**0.207**	0.196	177.33	**26.19**	✓
P5	**0.306**	0.224	**30.74**	33.36	✓
P6	0.146	**0.435**	240.55	**63.71**	✓
P7	0	**0.070**	289.73	**131.62**	✓
P8	**0.007**	0	**54.360**	195.17	✓
P9	**0.597**	0.550	302.81	**9.64**	✓
P10	0.433	**0.659**	181.74	**63.81**	✓
P11	0.416	**0.442**	355.98	**43.87**	✓
P12	**0.480**	0.445	233.78	**16.03**	✓
$\emptyset^*$	0.350 ± 0.18	0.410 ± 0.27			

TABLE 5.4: Achieved DSCs and sHDs on right femur for each test patient. $\emptyset^*$ depicts the mean over all patients, that actually have avascular necrosis on this side.

Table 5.4 shows the achieved DSCs and sHDs of both architectures on the right femur for all test patient volumes. In this case, all patients except for P3 have AVNFH on their right side. Removing P3 from the mean calculation $\emptyset^*$, the mean DSC values change to 0.350 for U-Net and 0.410 for the MTL-Net from the initial observations in Table 5.2. Like on the left side, it is visible, that the sHD improves drastically in the MTL-Net results compared to U-Net for most cases.

From the initial observations in Table 5.2 and the more differentiated inspections by means of Tables 5.3 and 5.4, it can be concluded that the proposed MTL-Net generally yields better extraction results for AVNFH than the U-Net baseline. This can also be observed in the exemplary comparison of segmentation outputs in Fig. 5.3 (b) and (d). It is noticeable that the segmentor branch output (Fig. 5.3 (c)) yields contours that are closer to the boundaries of the necrotic area. It, however, also contains some outlier predictions, especially at the bottom. This may be due to the spatial restriction of the reconstruction loss to the surrounding neighborhood in Eq.5.3. This term only punishes reconstruction errors, which occur within the topographical neighborhood. Therefore, deviations outside this area are not corrected, which can be seen in Fig.5.3 (g) at the bottom, where very high intensity values occur. Since the architecture has a joint encoder, this behavior may propagate towards the segmentor branch, resulting in the observed outliers. This is corrected in the final segmentation output (Fig. 5.3 (d)) by the multiplicative foreground emphasis by means of the topographical branch's output (Fig. 5.3 (f)). This yields a segmentation result for small

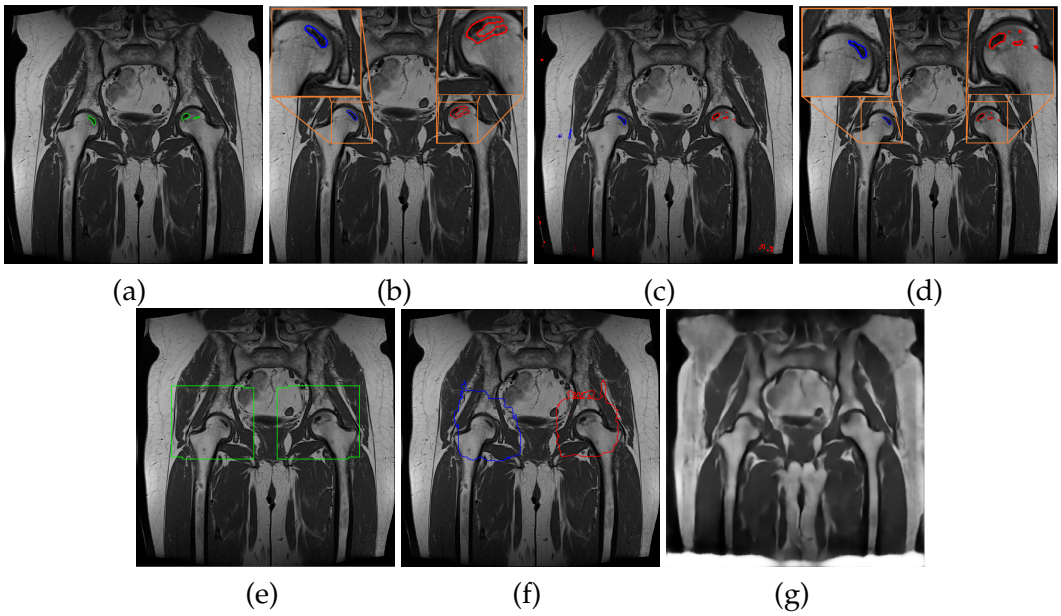


FIGURE 5.3: Exemplary coronal MRI slice with overlayed (a) ground truth, (b) U-Net output, (c) segmentor branch output, (d) final output, (e) relaxed ground truth, and (f) topographical branch output. The autoencoder output is depicted in (g). Based on [Pha+20].

structures, that is less prone to outlier predictions than a standard U-Net implementation.

5.1.5 Summary

In summary, a 2D multitask deep learning architecture for the segmentation of small structures is presented in this section on the example of AVNFH extraction. The reconstruction property of autoencoders is leveraged and a topographical localization objective is defined to improve the main task of segmenting AVNFH. Furthermore, a reconstruction-dependent adaptation scheme is applied during training. In the evaluation, promising improvements compared to U-Net can be observed, although the dice score needs to be regarded with care for these small structures.

5.2 Excursion: Topographical Priors for Deep Weakly Supervised Caries Localization in Smartphone Images

In this section, a related excursion to the application of topographical priors in a weakly supervised caries localization method is made, based on [Pha+21]. First, both medical and technical motivations are outlined in section 5.2.1, followed by a short literature review about related work in section 5.2.2. Then, a brief introduction to region-based convolutional neural networks (R-CNN) for fully supervised object detection is given in section 5.2.3. Afterwards, a weakly supervised approach is presented in section 5.2.3, that makes use of additional topographical priors during training for a better localization capability. In section 5.2.4 the conducted experiments are presented.

5.2.1 Motivation

Medical Motivation

While in developed countries routine dental consultations are often covered by insurance, access to prophylactic dental examinations is often expensive in developing countries. Therefore, sufficient oral health prevention, particularly early caries detection, is not accessible to many people in these countries, yet. Dental caries is a global oral health problem which can be effectively prevented and controlled through a combination of individual, community and professional efforts. Computer aided tools not only assist the dentists in accelerating the caries detection and diagnosis process, but may also be helpful in reducing human errors. Existing technical systems for the detection of tooth decay are based on standardized imaging techniques, such as x-ray. While access to this kind of technology is not always guaranteed in developing countries, smartphones have become available and affordable in most countries [May+16]. This circumstance can be utilized for affordable initial caries inspection to determine the necessity for a subsequent dental examination. Therefore, this section investigates the possibility of caries localization in smartphone images with both fully supervised and weakly supervised object detection methods.

Technical Motivation

Fully supervised object detection methods usually require the exact locations of the object of interest's bounding box for training. Therefore, a bounding box annotation is needed for each caries occurrence. Establishing these fully informative annotations (FIA) is often tedious and costly, particularly for large data sets, in which the target object may appear multiple times within one image. Weak supervision poses a more time-efficient but more challenging option to address object detection with less informative annotations. In context of this thesis, image-level annotations (ILA), in which only the image label but no information about the caries location is available, are considered as *weak* labels. Furthermore, mouth region annotations (MRA), in which a bounding box of the mouth region is given, is considered as additional topographical information. These are represented as binary rectangular masks.

Ren et al.'s Faster R-CNN [Ren+15] is used for the fully supervised upper baseline with FIAs. For the weakly supervised case, a CNN classifier is trained with the ILAs, where the activation maps within the CNN are utilized to locate possible caries occurrences. This is a challenging task as the caries regions may appear at multiple and different locations, varying scales, and under a variety of camera perspectives. The

initial weakly supervised strategy is compared to an extension with a topographically aware MRA based training approach.

5.2.2 Related Work

Primarily, the detection of dental caries has been a visual process, principally based on visual-tactile examination and radiographic examination [Sri+20]. Using these methods, caries can only be diagnosed by a dental health professional. Due to the asymptomatic initiation and progression of the dental caries, patients often fail to consult a dentist in time, resulting in dental caries progression to an irreversible loss of the dental hard tissue [Hum+19]. Ali et al. [AEZ16] and Choi et al. [CEK18] make use of neural networks to automatically detect caries areas within x-ray images. Casalegno et al. [Cas+19] make use of near-infrared transillumination imaging instead of x-ray to extract caries regions by means of a U-Net like [RFB15] deep learning architecture. Kositbowornchai et al. [Kos+06] also train a neural network to detect artificial dental caries using images from a charged coupled device (CCD) camera and intra-oral digital radiography. For their evaluation only teeth with artificial caries are considered, which usually have different properties than naturally affected ones. Similar to this contribution, Datta et al. [DC15] use RGB images to detect caries regions, however, by means of traditional image processing methods, i.e. image enhancement, transforming the images into a different color space and clustering in this color space. In contrast to the following use case. their images were captured with a specialized camera for the oral cavity, which allows similar lighting conditions and viewpoints across the data set. Saravanan et al. [SRG14] propose a strategy to detect dental caries in its early stage using histogram and power spectral analysis. In this method, the detection of tooth cavities is done based on the region of concentration of pixels with regard to the histogram and based on the magnitude values with regard to the spectrum. Zhang et al. [Zha+20b] and Liang et al. [Lia+20] propose deep learning-based localization systems for cavity detection and integrate their systems into smartphone applications. Liu et al. [Liu+19] explore the applicability of in-home dental healthcare by presenting a complete IoT system, in which deep learning is used for object localization. Most of these contributions have in common that either full supervision or traditional unsupervised image processing is applied to detect the caries region.

Regarding weakly supervised methods, related work can be generally categorized into multiple instance learning (MIL) based methods [BPT14; Wan+19], CNN based approaches [Oqu+15], and a combination of both [Li+16]. In MIL based approaches, an image is usually considered to be a *bag of instances*, where the object locations represent the instances. The aim is to learn a discriminative representation by means of the image-level annotations, which is then used to detect positive object instances in positive images. In this thesis, however, the focus is on a solely CNN based approach for weakly supervised object detection, utilizing additional prior topographical knowledge.

5.2.3 Methods

Fully Supervised Object Detection

For the fully supervised scenario with fully informative annotations (FIAs) of the caries' bounding boxes within each image, a variant of region based convolutional neural networks (R-CNNs), as introduced by Girshick et al. in 2014 [Gir+14] is used.

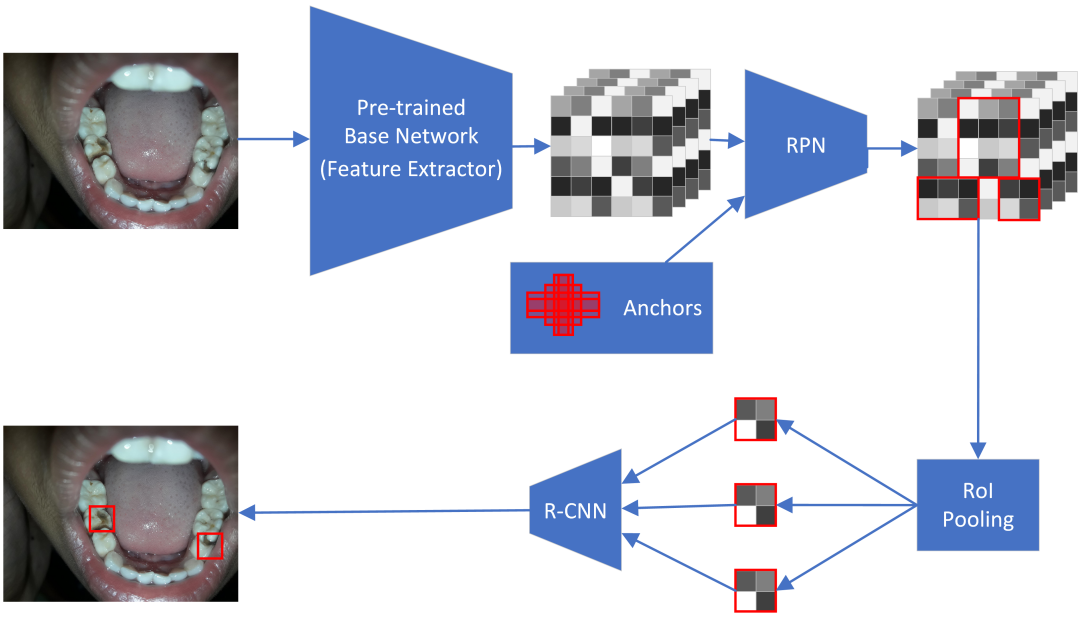


FIGURE 5.4: Scheme of the Faster R-CNN architecture, as proposed by Ren et al. [Ren+15]. Reused from [Pha+21].

The basic idea is to propose regions of interest (ROIs), i.e. possible object regions within the input image (e.g. by a selective search), which are then classified by a CNN to predict the class labels (including background) for the proposed ROIs. The concept of R-CNNs builds the foundation of various subsequent object detection algorithms.

Faster R-CNN

For the use case of caries detection in smartphone images, Ren et al.'s Faster R-CNN approach [Ren+15] is a suitable architecture. It is based on Girshick's work on Fast R-CNNs [Gir15], an improvement of the initial R-CNN proposal [Gir+14]. In contrast to these initial R-CNN iterations, Ren et al. formulate an end-to-end strategy, in which the region proposal is carried out by a CNN. The general pipeline is depicted in Fig. 5.4. First the input image is processed by a pretrained CNN, e.g. by Simonyan and Zisserman's VGG-16 [SZ15b] or He et al.'s ResNet-50 [He+16], to extract the image's latent features with their last convolutional layer. Then, a region proposal network (RPN) processes the feature map stack to propose possible ROIs. For this, the RPN learns to classify whether so called anchors, i.e. bounding boxes with specific size and aspect ratio for each image position, are possible object regions or not. Additionally, the RPN learns to refine the anchor boundaries. Since the proposed regions of the RPN may be of different size, a ROI pooling layer transforms the proposed regions to a uniform size. A ROI pooling layer is able to reduce a region of arbitrary size to a predefined size by using as many pools on the input region as the predefined size suggests. The uniformly sized ROIs are then passed to the R-CNN to be classified. A particular advantage of the Faster R-CNN approach is the end-to-end trainable architecture and the network's capability to locate even small objects, such as caries cavities.

Weakly Supervised Localization

For the weakly supervised case, only image-level annotations (ILAs) are utilized for training. The main idea is to train classification CNNs, which only require ILAs, and apply a variant of Zhou et al.'s class activation mappings (CAMs) [Zho+16]. While these are originally introduced to visually explain the decision processes of specifically designed classification CNNs, they are utilized to localize the image regions, which are responsible for the classification as caries. The underlying assumption is that specifically caries regions should be used by the CNN for its classification.
The general idea of CAMs is to inspect the influence of each feature map and their activations after the CNN's last convolutional layer. A limitation to Zhou et al.'s CAM is a restriction on the design of the CNN, as it needs to implement a global average pooling (GAP) layer after the last convolutional layer, followed by *one* fully connected (FC) layer. This is, however, a major restriction, as this CAM approach is not applicable to many classification networks, which do not use GAP. The importance of each feature map (after the CNN's last convolutional layer) is estimated according to the weights of the FC layer. The weighted sum of these feature maps results in a CAM for each class, visualizing which regions of the image are responsible for the class assignment.
For the caries localization task, the activated regions of the CAMs are treated as possible caries locations.

Gradient-weighted Class Activation Mapping

Due to the CNN design limitation of Zhou et al.'s CAMs, a more general extension of CAMs, presented in Selvaraju et al.'s work on *gradient-weighted class activation mappings* (Grad-CAMs) [Sel+17], is used instead.
Let F_k denote the k-th feature map of a feature map stack with $k \in \{0, 1, \ldots, K\}$, where $K + 1$ denotes the number of feature maps in that stack. The general idea is that the partial derivative of the model output $y^{(c)}$ for a class $c \in \mathscr{C}$ with respect to a feature map's pixel position (i, j), i.e.

$$\frac{\partial y^{(c)}}{\partial F_k(i,j)}$$

corresponds to the local influence for the class assignment to c. Therefore, the global average over the partial derivatives of $y^{(c)}$ with respect to all pixel positions of a feature map F_k determines the approximated influence of F_k on the class assignment of class c to the input image, i.e.:

$$\alpha_k^{(c)} := \frac{1}{M_k N_k} \sum_i \sum_j \frac{\partial y^{(c)}}{\partial F_k(i,j)}, \tag{5.5}$$

where $(M_k \times N_k)$ is the size of the feature map F_k. Selvaraju et al. [Sel+17] propose computing the Grad-CAM for a class c by means of the weighted sum over all feature maps, where the weights are determined by the influence in Eq. (5.5), followed by a ReLU activation, i.e.:

$$A_{Grad-CAM}^{(c)} := ReLU\left(\sum_k \alpha_k^{(c)} F_k\right). \tag{5.6}$$

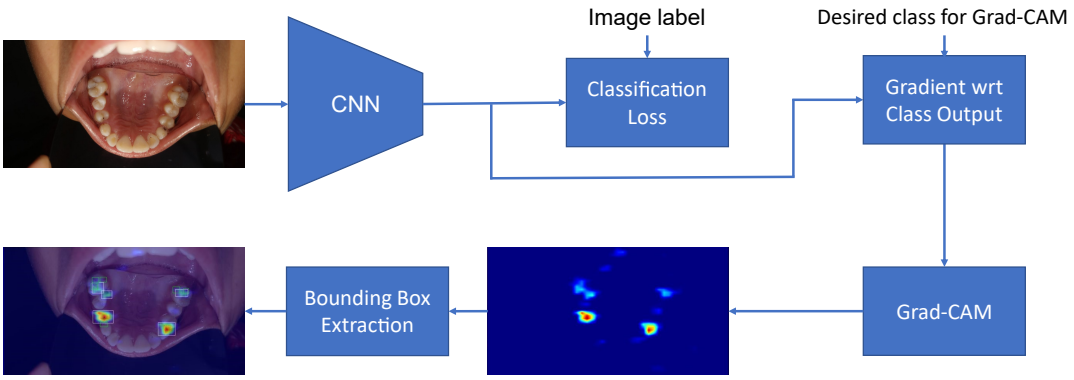


FIGURE 5.5: Proposed caries detection pipeline. Grad-CAM is only used for inference.

The ReLU activation helps to only consider regions, which have a positive impact on the assignment to the class c.

Proposed Caries Detection Pipeline

As mentioned before, the generated Grad-CAMs are used to extract possible caries areas, as they locate image regions which are responsible for the class prediction. First, a classification CNN, such as ResNet-50 [He+16], which is pretrained on ImageNet[Den+09a], is trained to differentiate between caries and non-caries images. For inference, the class activation map $A^{caries}_{Grad-CAM}$ for the caries class is generated by means of the gradients of the desired class output (in this case *caries*). Bounding boxes of possible caries locations are derived from the Grad-CAM. The overall pipeline is depicted in Fig. 5.5.

Fig. 5.6 elaborates on the generation of bounding boxes, based on the Grad-CAM pipeline. Otsu thresholding [Ots79] is applied on $A^{caries}_{Grad-CAM}$ to generate a binary mask, that only keeps relevant activation positions. By point-wise multiplication relevant activation locations are isolated. Afterwards, Gaussian blurring is applied to

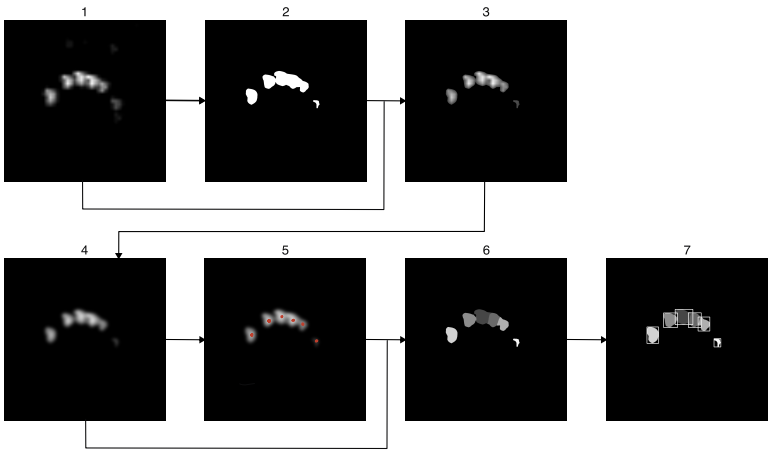


FIGURE 5.6: Bounding box extraction process from Grad-CAM. (1) Grad-CAM, (2) mask after Otsu thresholding, (3) Hadamard product, (4) smoothed by Gaussian blur, (5) extraction of local maxima, (6) random walk segmentation, (7) extracted bounding boxes. Reused from [Pha+21].

smooth the activation landscape for a subsequent extraction of local maxima. These are used as seed points for a random walk algorithm to segment the remaining image [Gra06]. Based on these segmentation results, the bounding box limits for predicted caries locations are finally estimated.

Extension with Topographical Constraints

To further improve the localization capabilities of the proposed pipeline, additional annotations of the mouth region (MRAs) are considered. These auxiliary location constraints are injected into the classification network only during training by extending the classification loss by a topographical localization constraint term $\mathscr{L}_{constraint}$. Let χ_{mouth} denote a binary mask, indicating the mouth region. Then the topographical loss for class $c \in \mathscr{C}$ is defined as

$$\mathscr{L}_{constraint} := \sum_{(i,j)} A^c_{Grad-CAM}(i,j) \cdot (1 - \chi_{mouth}(i,j)), \tag{5.7}$$

punishing any activation outside the mouth area. The pipeline is depicted in Fig. 5.7. In the previous proposal, the Grad-CAM is only computed during inference. In this topography aware procedure, however, the Grad-CAM and the location constraints are taken into account during training.

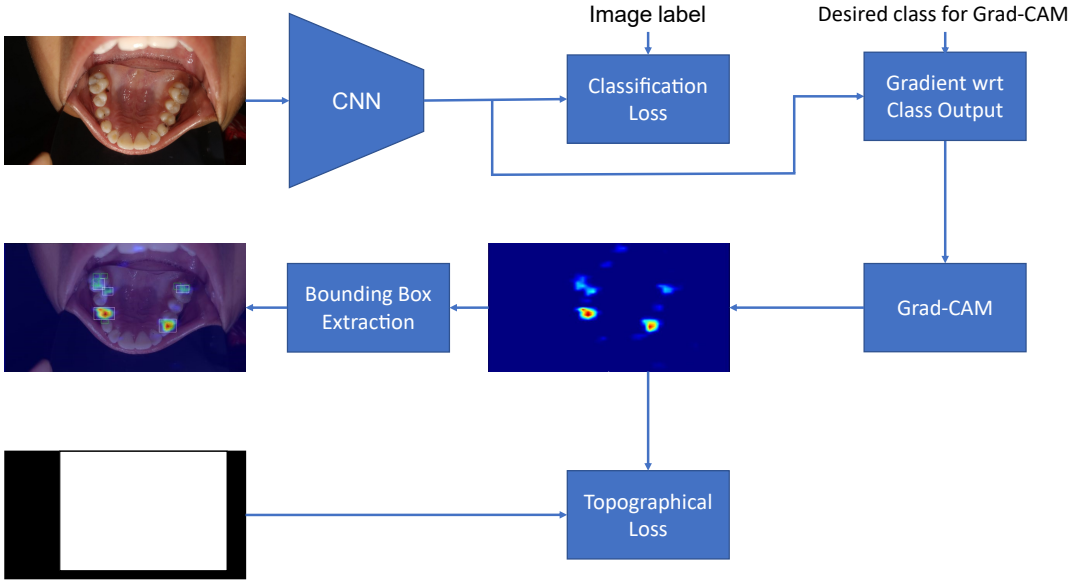


FIGURE 5.7: Proposed caries detection pipeline with topographical constraints. The topographical loss is combined with the classification loss to train the classification CNN. Here, Grad-CAM is also used during training to backpropagate from the topographical loss.

Implementation

For the Faster R-CNN implementation, as an upper baseline, the Tensorflow Object Detection API[1] is utilized. For feature extraction, ResNet-50 [He+16], pre-trained on ImageNet[Den+09a] with an input size of 1024×1024, is used as the base network. The RPN is empirically configured for anchors with scales of $0.25, 0.5, 1.0, 2.0$ and

[1] https://github.com/tensorflow/models/tree/master/research/object_detection

aspect ratios of $0.5, 1.0, 2.0$, resulting in 12 different anchor variations per location. For training a batch size of 16 is used. A momentum optimizer with cosine learning rate decay, initialized with a base rate of 0.1, is employed for weight adaptation. Due to the given implementation, a maximal number of training iteration steps of 15000 is set, instead of setting a maximal number of epochs. For the proposed weakly supervised strategies, ResNet-50 is also used as the base classifier network for comparability to Faster R-CNN. The methods are implemented in Tensorflow 2.0. The model is trained with a batch size of 8 for maximally 100 epochs, employing the RMSprop optimizer with a learning rate of 0.001.

5.2.4 Experiments

Data

For the experiments, a data set with annotations is provided by the Department of Paedodontics and Preventive Dentistry of UCMS and GTB Hospital, Delhi, India. It consists of 387 smartphone images of the oral cavity, from which 220 are of patients with caries and 167 images show healthy teeth. From the 220 caries images, for 93 images the exact caries locations are annotated with bounding boxes (FIAs), whereas the remaining images only have image-level annotations (ILAs). The mouth region annotations (MRAs) are created for all 387 images. For image collection, a OnePlus 7 pro smartphone composite camera[2] system was used, consisting of three different $(48, 16, 8)$ mega pixels cameras. All images were taken while considering the normal picture taking behavior of a user. For example, the camera is focused on the tooth cavities (if apparent) and pictures are taken in zoom out mode. For both patients with healthy and patients with caries affected teeth, the camera focus was set on the oral cavity. The patients were asked to keep still to avoid blurry images. Depending on the exposure of the patient to natural light, artificial light of the smartphone has been used to increase visibility if necessary. The field of view and the perspective of the images varies, depending on the position of the carious lesions.

The evaluation is conducted in a hold-out manner, in which the data set is divided into non-overlapping training, validation and testing sets with a ratio of approximately $60 : 20 : 20$. The training set is used for training, the validation set for monitoring, and the testing set for the evaluation. First, the classification performance of the base classifiers for the classification task is evaluated. Then, the caries localization performance is investigated. It needs to be noted that only caries images with FIAs (and all non-caries images without location information) can be used for training the Faster R-CNN approach. These are divided according to the aforementioned ratio.
Therefore, all testing images are used for the classification evaluation, whereas only testing images with FIAs (and all non-caries testing images) are considered for the object detection evaluation. The Faster R-CNN system is used as a baseline to estimate the upper bound for the localization task, as it is to be expected that a fully supervised system yields better results than weakly supervised approaches. In the following, the weakly supervised caries detection method is denoted as *WSCDM* and its extension with local constraints as *WSCDM-LC*. Both systems are trained with ILAs and WSCDM-LC additionally with MRAs.

[2] `https://oneplus.com/de/7pro#/specs`

Classification Performance

Since the weakly supervised strategy is based on a base classificaton CNN, it is crucial to analyze whether the base classifier yields reasonable results. To evaluate the classification performance of the Faster R-CNN, images for which caries locations are predicted are considered caries image classifications. Precision, recall, and DSC serve as evaluation metrics (see chapter 2.4).

Metric	Faster R-CNN	WSCDM	WSCDM-LC
Precision	0.61	1.00	1.00
Recall	0.90	0.76	0.86
DSC	0.73	0.86	0.92

TABLE 5.5: Classification results regarding precision, recall, and DSC of fully and weakly supervised systems on test set. From [Pha+21].

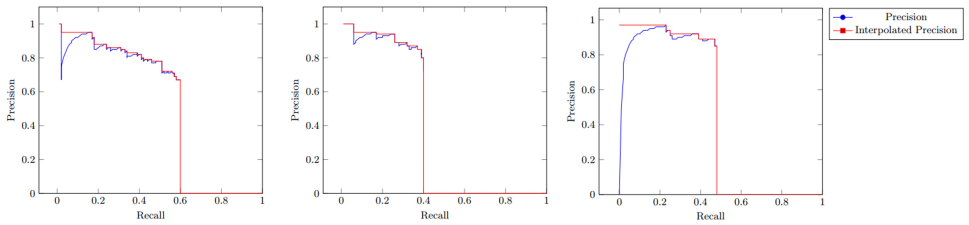
Table 5.5 shows the achieved results of the three systems on the unseen testing set in comparison. It is noticeable that both WSCDMs achieve a precision of 1, which means that both systems have predicted zero false positives on the test data set. This implies a generally low false positive rate for the base classifiers of the weakly supervised systems. However, Faster R-CNN yields the highest recall, which is favorable in terms of clinical applicability, as this implicates the lowest false negative rates, thus this system is less prone to oversee caries images. Regarding DSC it is observable that WSCDM-LC obtains the best results, at least for classification.

Localization Performance

For a comparison of the localization capabilities, the *mean average precision* (mAP) is used as the performance metric. To describe the mAP, first the computation of the *precision-recall curve* is elaborated.

Any bounding box prediction that has an intersection over union (IoU) with any ground truth annotation over a threshold is assumed to be a true positive (TP). Any prediction, for which an IoU over this threshold cannot be achieved with any available ground truth bounding box, is considered a false positive (FP).
For the evaluation, first *all* test samples are passed through the trained model to achieve bounding box predictions. An initially empty set is then iteratively filled by one of the bounding box predictions. The order, in which the set is filled, is determined by the prediction confidence, beginning with the highest confidence score. Here, the prediction confidence refers to the output score, that the model calculates for the class of the corresponding bounding box. In every iteration, precision and recall are calculated. For precision only the predictions within the set are considered, however, for recall all actual positives, also those outset the set, are taken into account.
For the *first* iteration, a very low recall value is expected, as the number of true positives in the set yields a small fraction of all actual positives. However, the precision is either 1 or 0, as all of the predictions within the set so far (i.e. the first and highest ranked prediction) are either correct or not. While the recall value can only rise or stay the same with each additionally inserted prediction, the precision value rises with correct and decreases with false additional prediction insertion.

For each network prediction a precision-recall pair is obtained by the above mentioned procedure, which can be visualized in a precision-recall curve. Usually the precision value of each recall is adapted, such that the resulting curve is monotonically decreasing with ascending recall. This is accomplished by using the maximally achieved precision of all *higher* recall scores. This is denoted as the *interpolated* precision. Selecting the maximally achieved precision of all *higher* recalls follows the intuition, that the higher the recall, the more difficult it is to achieve a high precision. If many actual positives are correctly classified as positive, the classifier could tend to classify actual negatives as false positives, which would result in a high recall, but a low precision. Thus, if for a high recall, a high precision is achieved, it can be assumed, that at least this precision score can be also achieved for lower recall values.



(a) Faster-RCNN (b) WSCDM (c) WSCDM-LC

FIGURE 5.8: Precision-Recall curves for IoU level of 0.5.
Reused from [Pha+21].

The resulting precision-recall curves of the experiments are shown in Fig. 5.8. The average precision (AP) approximates the area under the curve (AUC). Since this procedure can be done for multiple classes, the mAP is often defined to be the mean AP over all classes. In this particular caries detection case, AP and mAP are the same, as the number of classes to detect is one.

The precise definition of AP, however, varies depending on the context. For example, in the Pascal Visual Object Classes (VOC) challenge of 2007 [Eve+07; Eve+10b; Eve+15] the AUC, is approximated by averaging over eleven equidistantly selected precision values of the interpolated precision-recall curve, following Salton and McGill's suggestion [SM83]. In the Pascal VOC challenges from 2010 onward, on the other hand, the AP is calculated by integration, i.e. by a more precise calculation of the AUC [Eve+10a; Eve+15]. While in the Pascal VOC challenges an IoU threshold of 0.5 is used to distinguish between TP and FP, in the Common Objects in Context (COCO) challenges [Lin+14] a series of IoU thresholds is used to generate different precision-recall curves. The average over all resulting AUCs (calculated as in Pascal VOC 2010 by integration) is then considered the AP.
For all three variants, the mAP denotes the mean over all considered classes. In this thesis, the mAP definitions of the Pascal VOC [Eve+07; Eve+10a] and COCO challenges [Lin+14] are considered. Tab. 5.6 shows the various achieved mAP values for each system. For the COCO variant, the values after @ denote the considered IoU threshold(s). The formulation @$[0.5 : 0.05 : 0.95]$ refers to the usage of IoU thresholds in the range of 0.5 to 0.95 with a step size of 0.05. For a threshold of @0.5 the COCO mAP definition is identical to the Pascal VOC 2010 definition.

		Faster R-CNN	WSCDM	WSCDM-LC
Pascal VOC	2007	0.55	0.43	0.43
	2010	0.53	0.41	0.45
COCO	@$[0.5 : 0.05 : 0.95]$	0.46	0.29	0.34
	@0.75	0.51	0.38	0.45

TABLE 5.6: Detection results of caries locations on test set regarding Pascal VOC 2007 & 2012 and COCO mAP. Reused from [Pha+21].

The results in table 5.6 show that the fully supervised Faster R-CNN baseline approach yields the best caries localization results on the test data set across all considered mAP definitions. This outcome was to be expected, as the annotations contain more complex information than the image-level and mouth region annotations. However, the proposed weakly supervised strategies also achieve promising detection results, nevertheless. Except for the Pascal VOC 2007 case, the topography aware extension yields better results than the initial WSCDM for the remaining mAP definitions. The gap to the fully supervised Faster R-CNN upper baseline can be therefore reduced by the additional location constraints.

Fig. 5.9 shows the location predictions of each strategy on exemplary test images. The predicted bounding boxes are depicted as white, whereas the ground truth bounding boxes are shown in green. The activation maps of the weakly supervised systems are overlayed and the confidence score of the base classifier of the weakly supervised systems is depicted in the upper left corner. In the first row, it is observable that surprisingly the location prior, which is only used during training, seems to help in differentiating different instances, which WSCDM fails to achieve. The additional coarse location information possibly fosters some reallocation of available network resources.

Rows 2-4 depict example images, in which WSCDM-LC shows more intuitive activation maps focused within the mouth region, whereas in these cases WSCDM shows multiple activations in non-caries regions. This indicates a superior detection performance of WSCDM-LC compared to WSCDM, underlined by most mAP results in Table 5.6. In particular, row 3 shows an image, for which the base classifier of the WSCDM predicts a false negative (as the low prediction score of 0.1% for caries indicates), whereas the semantically more meaningful activation map of WSCDM-LC seems to help in an improved classification (prediction score of 96.44% for caries), which is also suggested in the overall superior classification results in Table 5.5.

Although row 1 suggests that the location constraint helps in differentiating adjacent instances, row 4 shows a drawback of the additional location constraint. For strong activations, the bounding box extraction strategy captures a larger area, as the neighborhood of a strong activation usually also shows strong activations. In this case the strong activation results in a bounding box prediction that is larger than the desired ground truth.

Row 5 shows an example test image, in which all systems, including Faster R-CNN, yield a false positive prediction in the bottom left area. A closer inspection shows, that all 3 systems have detected a caries region, which was not annotated in the

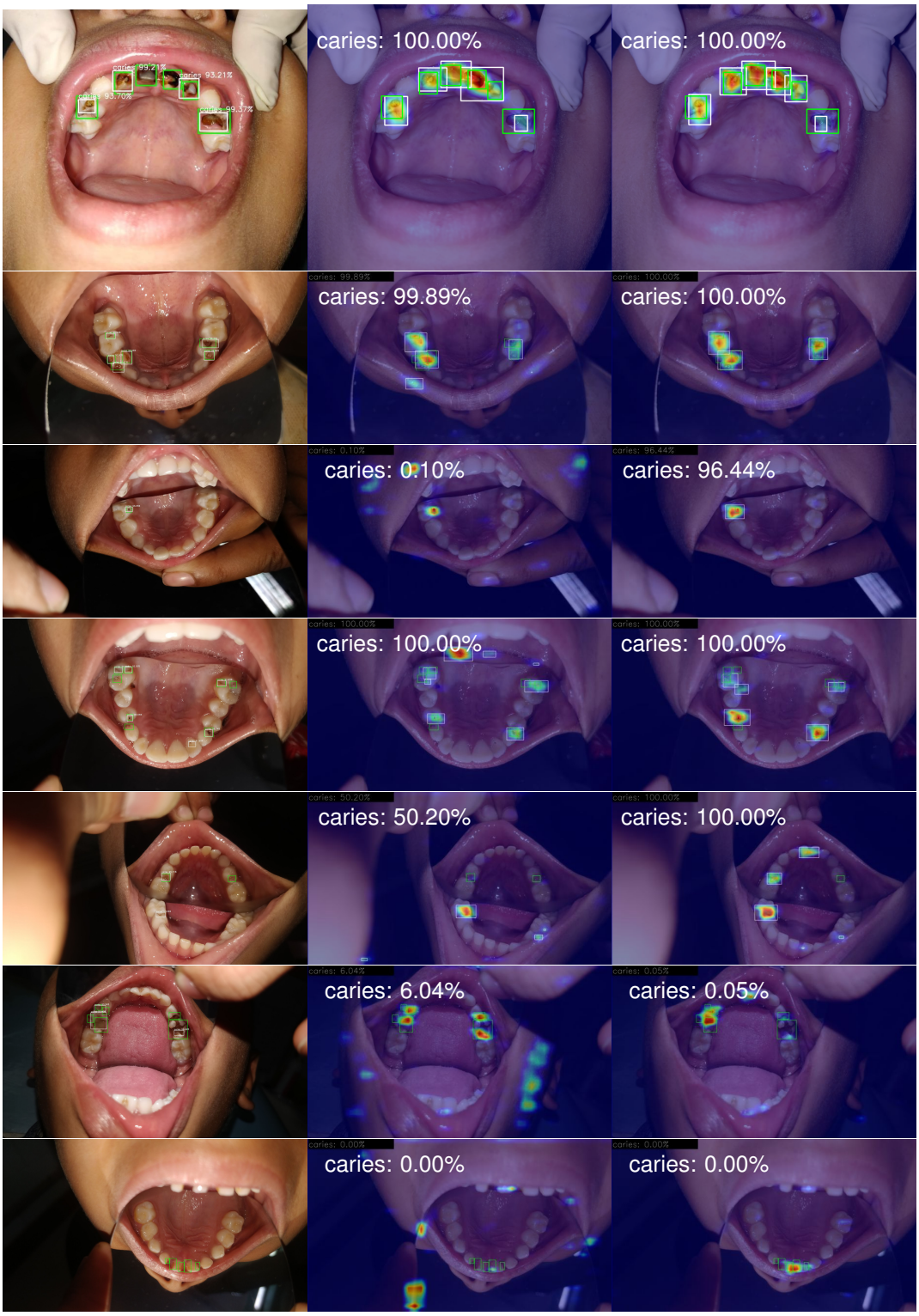


FIGURE 5.9: Exemplary predictions on test set. Ground truth bounding box in green, predictions in white. First row image was cropped to keep patient's anonymity. Reused from [Pha+21].

ground truth data, as the predicted region is labeled as caries in the dentist's mirror, but not on the actual tooth.

The last two rows show difficult examples, in which the base classifiers of the weakly supervised systems both predict false negatives. Therefore, no bounding boxes are generated. However, when inspecting the activation maps, it is still noteworthy, that WSCDM-LC shows more intuitive activation locations than WSCDM. Furthermore, it is noteworthy regarding the last row, that WSCDM-LC shows clear activations in very difficult caries areas, which even Faster R-CNN fails to detect.

5.2.5 Summary

In this section, an excursion to the task of weakly supervised caries detection using topographical priors is made. Both fully supervised and weakly supervised deep learning strategies are implemented to detect caries regions in smartphone camera images. For the weakly supervised case, a Grad-CAM based approach is proposed, in which the activation maps are used to locate the caries areas. This strategy is extended by incorporating topographical priors in terms of additional mouth region annotations during training.
It is demonstrated that both fully supervised and weakly supervised methods show promising detection results, although the gap between these approaches can only be partly reduced by means of mouth region annotations. Interestingly, the additional topographical priors not only concentrate the activations within the mouth region, but they also help the base network of the weakly supervised system towards better classifications. They also seem to improve differentiation between adjacent instances. Since CAMs have been mostly used for visually inspecting classification networks, that are often trained on images containing only one object, it is a promising observation, that multiple instances of the same class can be located, nevertheless. Additionally, the incorporation of topographical priors helps in differentiating adjacent instances.

5.3 Conclusion

In this chapter, a multitask learning approach is presented, using topographical priors to improve the extraction of small structures, such as avascular necrosis of the femoral head in MR volumes. Topographical constraints are introduced to the training process, in order to limit the range of feasible locations. The approach shows major improvements, particularly in the reduction of false positive outlier predictions.
Furthermore, a similar strategy is proposed in an excursion to the task of weakly supervised caries detection in smartphones, in which the additional topographical information during training generally helps in improving the detection quality.

Chapter 6

Application: Automated Necrosis Projection from MRI to X-ray

In this chapter, the proposed methods regarding shape priors (chapter 4) and topographical priors (chapter 5) are combined to address the practical task of necrosis projection from MR volumes to x-ray images.

6.1 Motivation

As described in chapter 5, avascular necrosis of the femoral head (AVNFH) can lead to functional damage of the hip joint. As a result, the hip joint may be irreversibly corrupted, if left untreated. When detected and treated in early stages, a complete hip joint replacement can often be prevented. Core decompression surgery is an indicated surgical procedure in early stages to reduce pressure, promote blood flow, and therefore encourage the healing process in affected regions, as mentioned by Wang et al. [Wan+17]. Here, a tunnel to the affected area is drilled and the necrotic bone tissue is then removed. In preoperative planning, MR images are often used for stage assessment, as necrotic tissue can be easily distinguished from healthy bone tissue. In surgery, on the other hand, (live) fluoroscopic x-ray images are used for navigation. However, necrotic tissue is not represented as well in x-ray images, particularly in early stages. Therefore, the surgeon often needs to manually project the necrotic area, found in the MR scan, to the x-ray images. In this chapter, a strategy is proposed, to automatically find necrotic area of the femoral head in MR volumes and project it onto a corresponding x-ray image.

6.2 Related Work

For a more precise assessment of the necrotic area, Li et al. [Li+18] describe a method which utilizes 3D prints of the femur and a patient specific guide plate, that is designed to aid in a more stable navigation of the drilling tools. For this, the femur and the necrotic area are extracted from CT scans during preoperative planning. Based on these extractions, a guide plate is modeled, such that it steers a drilling needle towards the necrotic area, when aligned correctly to the femur shaft during surgery. Cheng et al. [Che+20] on the other hand describe a procedure, in which the 3D printed guide plate is positioned on the patient's skin instead of directly on the bone surface.

In contrast to these methods, the following strategy does not require any additional 3D printers. Instead, the acquired MR volumes for preoperative planning and the initial fluoroscopic x-ray images, on which the necrotic area needs to be projected onto, suffice. A thorough search of the relevant literature did not yield any similar approach to improve the accuracy in positioning the drilling needle during surgery.

6.3 Methods

For the projection of necrotic area from MR volumes to x-ray images, a 3D to 2D registration strategy, based on *evolution strategies*, is proposed. First the femur is extracted from the fluoroscopic x-ray images, using a deep learning model. As the femur is a structure with only little variance in shape, especially in 2D images, the IE_2D-Net, presented in chapter 4.3 is a feasible architecture, leveraging this property. Then, femur and necrotic area need to be extracted from coronal MR volumes, rendering a 3D model for both structures. For this task, the multitask learning approach, using topographical information about the necrotic tissue's whereabouts, presented in chapter 5.1, is a predestined candidate for necrosis extraction. The femur extraction from the MR volumes is conducted in a slice-wise manner by U-Net, as concluded in chapter 4.6. Using the 2D and 3D extractions of the femur, a transformation needs to be found, that projects the 3D femur model to a 2D plane, such that the projection is aligned to the 2D femur segmentation from the fluoroscopic x-ray image. This transformation can finally be used to project the 3D model of the necrotic area onto the fluoroscopic x-ray image. Fig. 6.1 depicts the overall strategy.

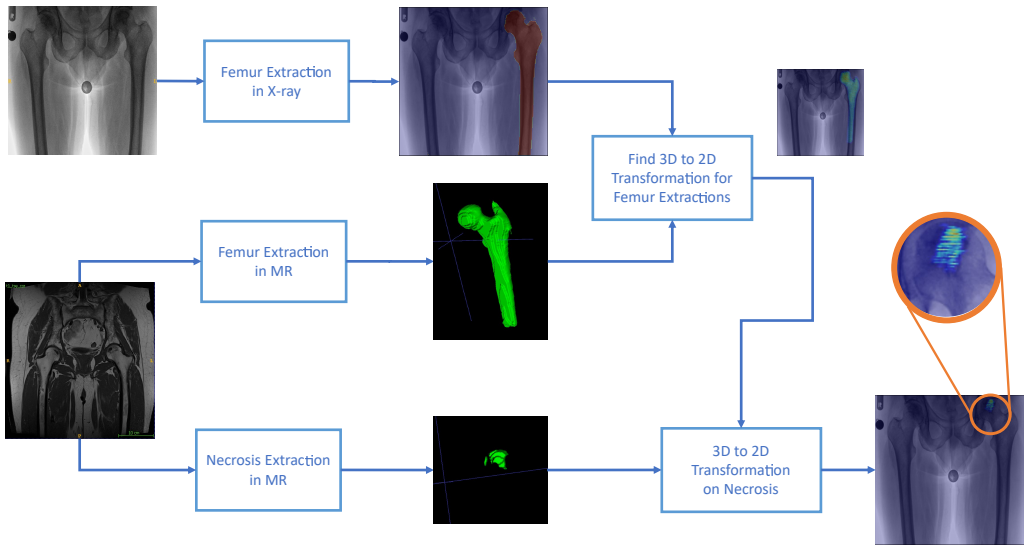


FIGURE 6.1: Overall strategy for necrosis projection from MRI to X-ray. In this approach the femur extractions in 3D and 2D are used to estimate a feasible transformation for the 3D extraction of necrotic tissue.

Since the extraction approaches have already been discussed, the estimation of a 3D to 2D projection remains an open topic to be addressed within this chapter. Basically, this is a registration task, in which the 3D femur model can be considered a moving image I_m and the 2D extraction a fixed image I_f. In accordance to chapter 2.2.1, a

transformation

$$T : \mathcal{I}_{3D} \rightarrow \mathcal{I}_{2D},$$

needs to be found, where $\mathcal{I}_{3D}$ denotes the space of all 3D images and $\mathcal{I}_{2D}$ that of all 2D images. The transformation T is acquired by minimizing a cost term $\mathscr{L}(I_f, T(I_m|\Theta))$, which considers the differences of the transformed 3D femur extraction $T(I_m|\Theta)$ to the fixed 2D femur extraction I_m, given the transformation parameters Θ, i.e.

$$\Theta^* := \underset{\Theta}{\operatorname{argmin}}\{\mathscr{L}(I_f, T(I_m|\Theta))\}$$

For the minimization of the loss term, evolution strategies, a specialization of *evolutionary algorithms*, are employed.

6.3.1 Evolutionary Algorithm

Evolutionary algorithms are a family of optimization heuristics, that follow Darwin's evolution theory of the survival of the fittest. The idea of implementing nature's selection mechanism dates back to the 1940's [ES+03] to Alan Turing's [Tur48] mentioning of "genetical or evolutionary search" in the context of achieving machine intelligence [Teu12]. Over time, practical implementations based on the same principal idea followed partly independent of each other [Teu12] by Bremermann [Bre62], Fogel et al.[FOW66; Fog98], Holland [Hol73; Hol92], Rechenberg [Rec65; Rec73; Rec78], Schwefel [Sch77; Sch93], and Koza [Koz94]. For a more detailed introduction into the field of evolutionary algorithms and its subareas, coined by the aforementioned authors, the reader is referred to Eiben and Smith's *Introduction to Evolutionary Computing* [ES+03].

Following the terminology of evolutionary algorithms, any parameter configuration is represented by an *individual*. A set of configurations is represented by a *population*, consisting of multiple individuals. The general idea is to start with a random population that evolves to a *strong* population, measured by a selection criterion. The variation of the gene pool, i.e. the exploration of new configurations, is accomplished by means of reproduction and mutation. The adaptation towards an environment, i.e. the exploitation of existing *good* configurations according to the defined selection criterion, is realized by the selection of the *fittest*. Therefore, a quantitative *fitness* measure needs to be defined to assess the suitability of each individual.

Fig. 6.2 shows the general evolutionary algorithm cycle. Here, each individual is represented by a colored line of different length, depicting the different adaptable properties an individual may have. After each cycle a new *generation* of individuals is generated, in which the best individuals from previous generations may be conserved. The production of new individuals is accomplished by means of selection, reproduction, and mutation. Note that the preservation of surviving individuals of previous generations and the exact behavior of recombination and mutation may be adapted depending on the use case. In the scope of the underlying task, *evolution strategies*, introduced by Rechenberg [Rec65; Rec73; Rec78] and Schwefel [Sch77; Sch93], are used, which are a specialization of evolutionary algorithms. They particularly allow the parameters, that need to be adapted, to be real valued, which is a necessary condition for this task.

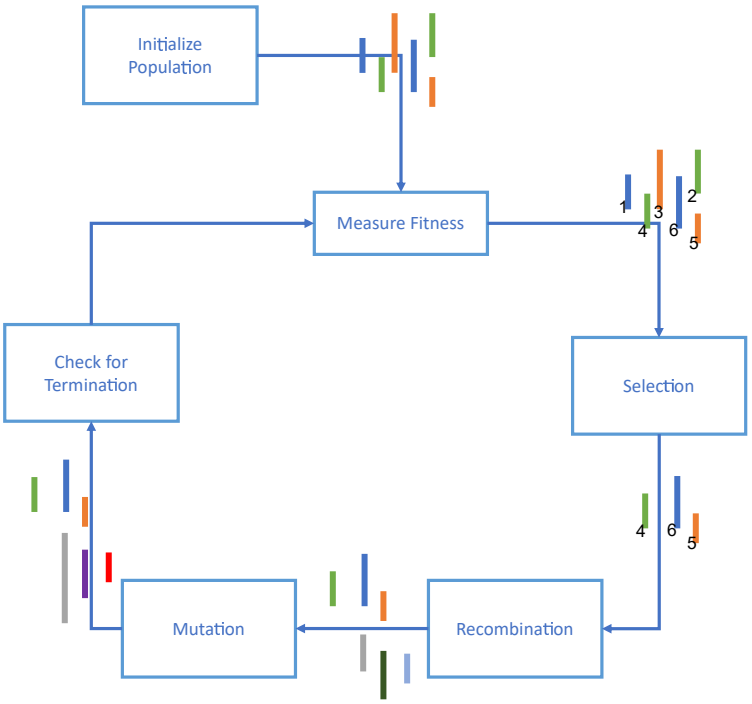


FIGURE 6.2: General concept of Evolutionary Algorithms. The adaptable parameters in this visual example are length and color. Although in this example mutation only affects the child generation in one parameter (length *or* color), the exact definition of mutation may be varied, depending on the use case.

Representation of Individuals

The transformation T is assumed to be *rigid*, comprising 3D rotation, translation, scaling, and parallel projection. The rotation can be realized by a rotation matrix R in the *Yaw-Pitch-Roll* [CK11] representation

$$R := \begin{pmatrix} \cos(r_y)\cos(r_z) & \sin(r_x)\sin(r_y)\cos(r_z) - \cos(r_x)\sin(r_z) & \cos(r_x)\sin(r_y)\cos(r_z) + \sin(r_x)\sin(r_z) \\ \cos(r_y)\sin(r_z) & \sin(r_x)\sin(r_y)\sin(r_z) + \cos(r_x)\cos(r_z) & \cos(r_x)\sin(r_y)\sin(r_z) - \sin(r_x)\cos(r_z) \\ -\sin(r_y) & \sin(r_x)\cos(r_y) & \cos(r_x)\sin(r_y) \end{pmatrix},$$

where r_x, r_y, r_z denote the rotation angles along the x-,y-, and z-axis, respectively. Scaling can be accomplished by the scaling matrix

$$S := \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix},$$

where s_x, s_y, s_z denote the scaling factors along the x-,y-, and z-axis, respectively. The translation is realized by means of a translation vector

$$\vec{t} := \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix},$$

where t_x, t_y, t_z represent the translation along the x-,y-, and z-axis, respectively. The transformations R, S, and $\vec{t}$ operate on the index space of the 3D femur segmentation map. After application of these transformations, the parallel projection is simply

accomplished by summing up of the binary values of the transformed segmentation map along the z-axis. Thus, further projection parameters are not required. All in all, the parameters that need to be adapted can be summarized as

$$\Theta := (r_x, r_y, r_z, s_x, s_y, s_z, t_x, t_y, t_z),$$

which yields the representation of an individual.

Selection

The goal of the selection process is to reduce the number of individuals within a population, making space for new individuals and therefore more parameter configuration variants. Each individual is rated according to a *fitness* function and only the fittest individuals *survive*, while the remaining individuals are removed from the population. By keeping the fittest individuals, they are conserved across generations. Regarding the underlying task, the fitness function needs to measure the alignment between the projection of the transformed 3D femur extraction to the fixed 2D extraction, given the parameters, that an individual represents. One possibility is using the dice score (see chapter 2.4.1) between the projection and the 2D extraction as fitness function. As mentioned before, the projection of the transform is accomplished, by summing up the tensor values of the transformed 3D segmentation along the z-axis. Any position with a value greater than zero is considered a projected femur point. However, for this variant the whole 3D femur needs to be transformed for each individual in each evolution cycle, which is computationally expensive and time consuming, particularly because many individuals may not represent promising parameter configurations.
Alternatively, the distance of *key features of the femur* to each other in the projection and the 2D segmentation can be measured, e.g. by the L_2 norm. For this, however, the key features need to be designed and extracted first, which is discussed in the subsequent subsection 6.3.2. Once the key features are set, they only need to be extracted once for the initial 3D and 2D femur segmentations. The transformation is then applied on the 3D key features instead of the whole 3D segmentation, which significantly reduces the computational workload. To measure the distance of the transformed 3D key points to the 2D key points, the z-coordinate of the transformed 3D points is neglected.

Recombination

To replace the eliminated individuals, new child individuals are generated from random pairs of surviving parent individuals. Let $\Theta_i^{(0)}$ and $\Theta_i^{(1)}$ denote the i-th component of arbitrary parents $\Theta^{(0)}$ and $\Theta^{(1)}$, respectively. Then the child configuration $\Theta^{(c)}$ is generated by means of the component-wise recombination

$$\Theta_i^{(c)} := \alpha\Theta_i^{(0)} + (1-\alpha)\Theta_i^{(1)}$$

for $0 < \alpha < 1$. Additionally, the population can be replenished with new random individuals.

Mutation

In the proposed evolution strategy, mutation only takes place on newly generated child individuals, to preserve the fittest individuals in the population. For the mutation process, Gaussian noise is added to each component of the individual. The mutation rate is steered by the chosen standard deviation for the Gaussian offset.

Termination

To terminate the algorithm, a maximal number of cycles can be defined. Additionally, termination can be enforced, as soon as an individual has achieved a minimally required fitness score or if the best individual does not improve for a certain number of cycles.

6.3.2 Key Feature Extraction

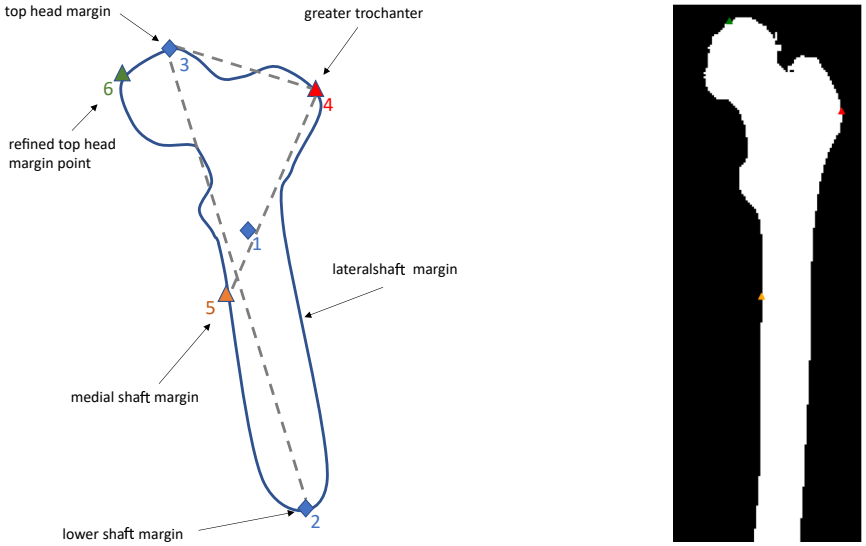


FIGURE 6.3: Illustration of key feature point extraction. In (a) the extraction steps are depicted with the used nomenclature, also showing the aiding connection lines. In (b) the extracted key points are depicted on an exemplary segmentation result from IE$_2$D-Net.

For a fast fitness estimation of an individual, a feature based fitness function is proposed. This reduces the computational workload of having to transform every 3D point of the 3D femur extraction to only a few key feature points. In particular, a six step strategy to extract 3 key feature points is presented, that makes use of the femur's shape properties. The following procedure, which is also illustrated by Fig. 6.3 (a) with the used nomenclature, is applicable for both 3D femur extraction as well as 2D femur extractions in an anterior-posterior view.

1. First, the centroid is calculated by averaging over all points that have been classified as femur. Because the distal femur mainly consists of the shaft, whereas the proximal femur comprises the larger femur head and collum, the centroid is expectedly closer to the proximal femur.

2. The lower shaft margin point $p_{low-shaft}$ of the femur segmentation is estimated as the furthest margin point from the centroid.

3. Consequently, the top femur head margin $p_{top-head}$ can be assumed to be the furthest point from the lower shaft margin $p_{low-shaft}$.

4. The greater trochanter $p_{troch-maj}$ is the **first key point** to be extracted. It can be estimated by using the furthest margin point from a line, drawn between the top femur head margin point $p_{top-head}$ and the lower femur shaft margin point $p_{low-shaft}$.

5. The distance between the greater trochanter $p_{troch-maj}$ and the top head margin $p_{top-head}$, denoted as d_{t-h} is used as a size indicator, which is independent of the shaft length. This is particularly important, since the shaft length may drastically vary depending on the field of view of the considered x-ray image. The **second key point** is achieved by selecting the margin point that both has a distance of $1.2 \cdot d_{t-h}$ from the connecting line between the top head margin $p_{top-head}$ and the greater trochanter $p_{trochmaj}$ and is additionally closest to $p_{top-head}$. The second condition ensures that the margin point is on the medial shaft margin, instead of the lateral one. This key point is denoted as the medial shaft point $p_{med-shaft}$. The factor 1.2 is estimated experimentally.

6. The **third key point** is a refined femur head margin point $p_{ref-head}$. This point is achieved by taking the furthest margin point from the line connecting $p_{med-shaft}$ and $p_{troch-maj}$, and which is closest to $p_{top-head}$. The second condition ensures that the third key point is in fact on the femur head margin, instead of the femur shaft margin.

In Fig. 6.3 (b) the three extracted key feature points are shown on an example 2D segmentation map.
Phan and Ko [PK15] focus on similar landmarks of the proximal femur, specifically the greater trochanter (step 3) and the fovea of the femoral head, which is located close to the proposed femur head margin point (step 6). The greater trochanter is also mentioned by Fischer et al. [Fis+20] and Gharenazifam and Arbabi [GA14] as a landmark of interest regarding the proximal femur. However, instead of extracting landmarks for subsequent registration, the aim in these publications is to achieve the aforementioned key points and further landmarks for femur specific measurements.

6.4 Experiments

6.4.1 Data

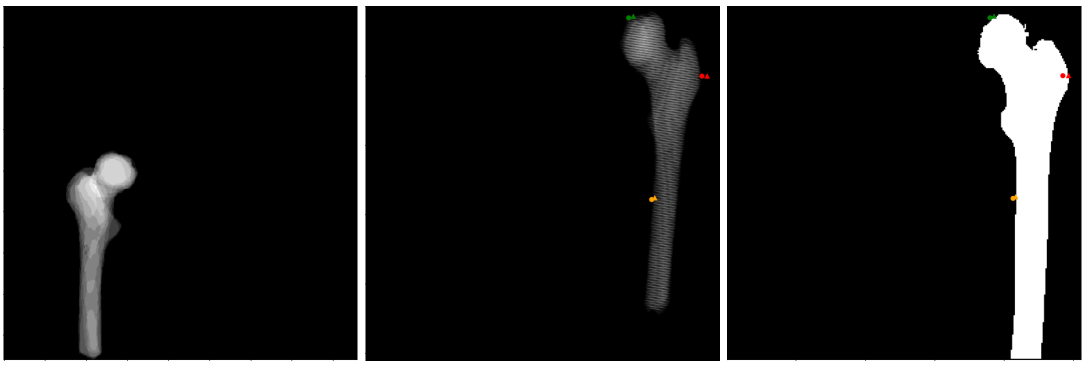
For the evaluation, the MR data from chapter 5.1 is used for 3D femur and necrosis extraction, and the x-ray data from the ablation studies from chapter 4.4.5 is used for 2D femur extraction. From all available data sets, there are eleven patients that have both X-ray and MRI scans, that can be used for this projection task. As concluded in chapter 4.6, U-Net is a suitable architecture for a slice-wise femur extraction from the MR volumes. For the 2D femur extractions, the predictions of an IE_2D implementation, presented in chapter 4.3, are used, as it appears to yield the most promising results in a standard segmentation setting, where the shape of the structure of interest shows only little variance and where the aspects of domain adaptation, few shot segmentation, and multi modal training can be neglected. For the extraction of the necrotic area, the predictions from the multi-task learning model from chapter 5.1 are used. For every prediction, may it be 3D or 2D, a leave-one-out strategy is used, i.e. the test patient image is held out, while the remaining available images are used

for training and validation. This yields eleven held out test patients, which are used for the subsequent registration method.

For a more meaningful investigation of the proposed projection strategy, the number of registrations can be artificially increased from 11 to 121 projections by pairing up each 3D segmentation of the eleven available patients with every 2D segmentation prediction, instead of only considering 3D/2D segmentation pairs which belong to the same test patient. This is a feasible procedure, since it can be assumed that the field of view may drastically diverge for 3D and 2D images of the same patient, one showing the whole lower body and the other one only the proximal femur. To estimate the projection quality, the dice score of the projection and the x-ray segmentation ground truth are used. The Hausdorff distance is not applicable in this scenario, since the field of view of the MR image may be different to the field of view in the x-ray image, which would lead to larger Hausdorff distances, even if the alignment is feasible. While the 3D predictions yield left and right structures, the predictions for the 2D x-ray images only contain the femur side which is considered for surgery. Therefore, only the corresponding side of the 3D extraction is considered for registration, which is also available in the 2D prediction.

6.4.2 Results

Since the 3D segmentation is used *as is*, no further information of the orientation of the coordinate system in the real world is utilized. This makes the registration task more challenging.



(a) (b) (c)

FIGURE 6.4: Visual projection comparison of initial femur position and after transformation. (a) shows the the initial femur projection onto the 2D coordinate system. (b) shows the projection after applying the estimated transformation. (c) shows the predicted 2D segmentation, that is used as fixed image. In (b) and (c) the extracted key feature points are also depicted.

In Fig. 6.4 (a) the initial projection of the 3D femur segmentation map onto the coordinate system of the 2D segmentation map is illustrated. The projection is accomplished by summing up the tensor values along the z-axis. The brightness therefore represents the thickness of the femur. It can be observed that the image coordinate systems of moving and fixed images are not aligned. Therefore, the projection of the left 3D femur appears on the left side (from the viewers perspective), whereas the left 2D femur extraction is on the right side. Fig. 6.4 (b) shows the projection of

the 3D femur segmentation, after applying the estimated transformation by means of the key point based evolutionary registration method. Additionally, the extracted key feature points of the 2D femur segmentation (triangles) and the transformed key feature points of the 3D segmentation (circles) are depicted to illustrate their distances after applying the proposed evolution strategy. Fig. 6.4 (c) shows the fixed 2D femur segmentation and the key feature points as a reference.

The resulting dice scores (DSCs) of the evolutionary registration method, using the proposed three key feature points, are shown in Tab. 6.1 in detail. An average DSC

2D \ 3D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	∅	std
P1	0.734	0.766	0.641	0.635	0.723	0.680	0.630	0.741	0.694	0.676	0.779	0.707	0.044
P2	0.816	0.841	0.850	0.770	0.688	0.739	0.672	0.668	0.784	0.740	0.779	0.768	0.052
P3	0.718	0.690	0.787	0.752	0.869	0.716	0.775	0.898	0.834	0.843	0.932	0.804	0.067
P4	0.888	0.880	0.890	0.845	0.919	0.927	0.812	0.900	0.912	0.831	0.832	0.882	0.034
P5	0.769	0.851	0.863	0.867	0.887	0.784	0.762	0.677	0.812	0.761	0.873	0.814	0.054
P6	0.743	0.863	0.876	0.796	0.892	0.927	0.809	0.926	0.794	0.886	0.947	0.865	0.054
P7	0.788	0.880	0.819	0.810	0.853	0.861	0.716	0.903	0.816	0.773	0.852	0.836	0.041
P8	0.765	0.864	0.830	0.854	0.882	0.791	0.784	0.762	0.820	0.834	0.882	0.828	0.036
P9	0.848	0.671	0.725	0.905	0.862	0.832	0.811	0.762	0.639	0.885	0.795	0.792	0.069
P10	0.828	0.901	0.776	0.758	0.662	0.815	0.669	0.855	0.825	0.741	0.810	0.797	0.058
P11	0.658	0.871	0.727	0.753	0.641	0.735	0.631	0.648	0.783	0.654	0.727	0.720	0.059
∅	0.778	0.825	0.799	0.795	0.807	0.801	0.734	0.795	0.792	0.784	0.837	0.801	0.068
std	0.051	0.063	0.061	0.056	0.093	0.065	0.064	0.093	0.049	0.065	0.055		

TABLE 6.1: Resulting DSCs from registration using the proposed key feature points. Columns denote the patient, from which the 3D femur extraction is used as moving image. Rows depict the patient, from which the 2D femur extraction is used as fixed image.

of 0.801 ± 0.068 over all registration combinations is achieved, which is a promising result. Especially the x-ray image of patient $P4$ yields a very good mean DSC of 0.882 ± 0.034, implying that it is particularly suitable for this registration method, as it shows robustness against the used 3D moving images. Interestingly, the best DSC scores are not always achieved along the diagonal of table 6.1, which one might assume, since here 3D and 2D images are from the same patient. As mentioned before, a possible explanation is a difference in field of view between 3D and 2D images, which renders the registration a more challenging problem for these cases. It also needs to be considered that DSC only reflects the overlap of the projected transformed segmentation with the 2D extraction of the *femur*, which only partly reflects the projection quality of the the necrotic area. Additionally, the DSC can only encapsulate the transformation quality to a certain extent. For example, a good alignment of a moving image with short femur shaft to a fixed image with long femur shaft would result in a worse DSC because of the missing femur shaft in the moving image. Nevertheless, the good overall DSC performance implies clinically usable necrosis projections.
Fig. 6.5 shows examples, in which the 3D to 2D femur registration succeeds in finding a feasible necrosis projection. The first row shows the femur projections, overlayed on the targeted x-ray images. The projections are depicted with a jet color map which indicates the thickness of the femur. Therefore, the red areas should not be confused with necrotic tissue. The second row accordingly shows the necrosis projections on the target x-ray images.

Fig. 6.6 illustrates the limitations of the proposed strategy. The first row shows the

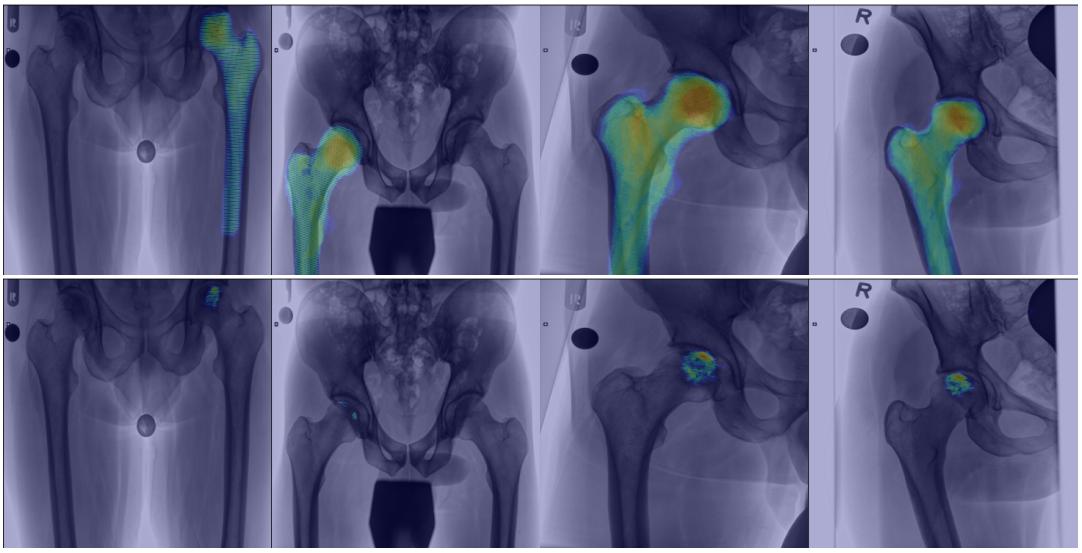


FIGURE 6.5: Exemplary good qualitative results of the proposed necrosis projection strategy. A jet colormap is used to visualize the thickness of the projected 3D structures, blue representing a thin and red a thicker region. Particularly for the femur projections, red areas should not be confused with necrotic tissue.

femur projections, while the second row depicts the necrosis projections. Although the femur projections are at the correct positions and show a feasible orientation, some shortcomings become apparent, nevertheless. The first image shows a case, in which the proximal femur is well aligned, whereas the femur shaft is not. The second and third images show examples, in which the femur head of the transformation is too large, rendering a necrosis projection outside the actual femur head in the x-ray image. The fourth example depicts a problematic scenario, in which the field of view of the moving 3D image is much smaller than the field of view of the fixed 2D image, which is visible in the large difference in femur shaft length.

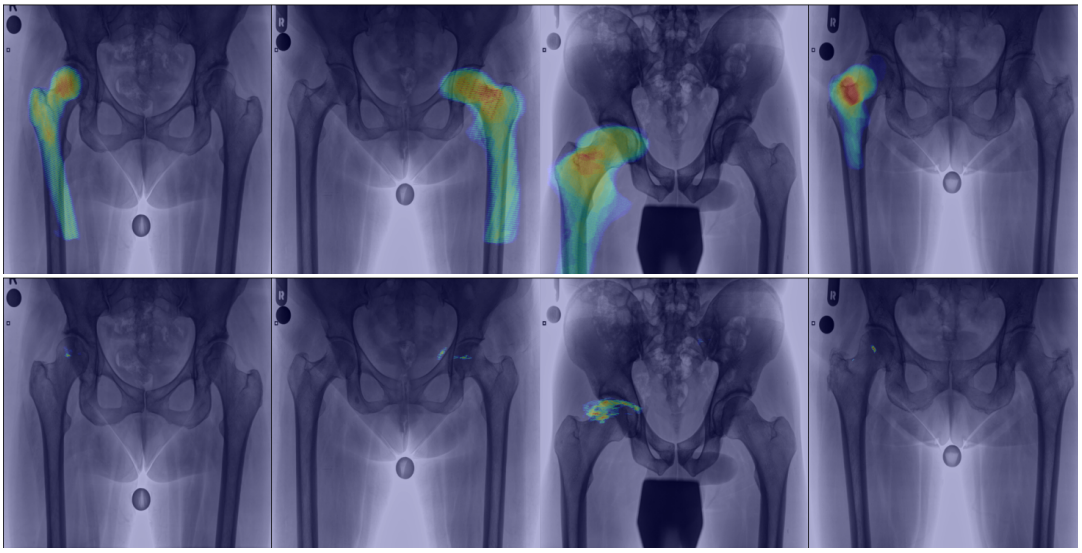


FIGURE 6.6: Exemplary bad qualitative results of the proposed necrosis projection strategy. A jet colormap is used to visualize the thickness of the projected 3D structures. Particularly for the femur projections, red areas should not be confused with necrotic tissue.

6.5 Conclusion

In this chapter, the task of necrosis projection from MR volumes to x-ray images is addressed. The concepts and conclusions of chapters 4 and 5 are combined for the extraction of necrotic tissue from MR volumes, and the segmentation of the femur in both 2D x-ray images and 3D MR volumes. To estimate the final projection transformation, an evolution strategy is proposed, that aims at projecting the 3D femur extraction onto the 2D femur extraction in its image coordinate system. For this, a key feature point extraction procedure is presented, to speed up the evolution process. The projection pipeline yields both promising quantitative and qualitative results. A limiting factor is, however, the small amount of key feature points, which are all located at the proximal femur due to the strong variance in FOV. This renders the correct positioning of the femur shaft difficult for some instances. Altogether, the proposed pipeline nevertheless represents an important contribution towards computer assisted intervention, incorporating the insights of the previous chapters.

Chapter 7

Summary and Outlook

This final chapter concludes this dissertation with a summarizing discussion of the presented approaches, which incorporate anatomical priors for fully automated segmentation pipelines, and gives suggestions for future research directions.

7.1 Traditional Image Segmentation

The presented initialization methods of chapter 3, namely Polar Appearance Models (PAMs) and Gradient based Expanding Spherical Appearance Models (GESAMs), demonstrate a sufficient initial contour estimation for subsequent contour based segmentation methods. Therefore, they are crucial components to complement traditional segmentation pipelines towards full automation. The retrospective hyper parameter analyses show for both suggested approaches that the learning based component is of crucial importance. However, they also reveal that some of the proposed cost function components could be neglected. Furthermore, the application of these methods is limited to single sided femur segmentation pipelines. In future research, a more general meta-framework would be desirable, which is capable of automatically weighting task relevant cost function components. A reduction of arbitrary structures with consistent shape appearance to suitable primitive shapes is a further aspect to be considered for more general initialization methods.

7.2 Deep Learning based Image Segmentation

In the context of deep learning, full automation is already implied by the end-to-end architecture design of fully convolutional segmentation networks. Therefore, the incorporation of anatomical priors is intended to improve the segmentation performance, instead of complementing deep learning methods towards full automation.

The proposed cascaded convolutional distance transform (CDT) from chapter 4.2 shows promising improvements in general segmentation settings, particularly regarding the symmetric Hausdorff distance, as shown in the example of thoracic organ segmentation. However, the transform is currently limited to two-dimensional distance transforms. Assuming a sufficient GPU memory capacity, an extension to a three-dimensional CDT could improve these initial results. While in scope of this

dissertation, the distance transform is used for shape characterization by the distance from the organ's own boundaries, an alternative strategy could be the characterization of an organ by the estimated distance to its neighboring organs. Furthermore, these initial results using a well established form of shape representation motivate for further combinations of deep learning methods with proven non-deep learning concepts, just like the distance transform.

The ablations studies of IRE$_3$D-Net, introduced in chapter 4.4, reveal that the IE$_2$D-Net from chapter 4.3 shows most promising and significant improvements compared to U-Net in a general segmentation setting. The application of IE$_2$D-Net (and shape prior based architectures in general) is particularly reasonable in segmentation settings where the structure of interest shows consistent shape appearance across all samples, even for varying FOVs.
In scope of the CHAOS challenge [Kav+21], the submitted IRE$_3$D-Net shows promising results, especially in cross-modality learning tasks, in which the network shows only little variation in the achieved scores across all test volumes. Because of the challenge character of the evaluation, it is however difficult to asses whether superiority is achieved through architecture design, hyper parameter tuning, or even advantageous computing resources. Therefore a direct comparison of the submitted architectures needs to be conducted with caution, although it is noteworthy, that the submitted IRE$_3$D-Net achieves superior results in the cross-modality learning tasks, even when trained on smaller scales to fit GPU memory.

In the one-shot setting, discussed in chapter 4.4.4, both IE$_2$D-Net and IRE$_3$D-Net show significantly improved results compared to a U-Net, which is trained in a comparable manner. One surprising observation is that U-Net already performs well, if training and testing volume have a large overlap in the ground truth mappings. The extension to one-shot multi-organ segmentation has not been explored in the scope of this dissertation and is certainly an important task for further research.
In general, the extension of imitating encoder based architectures towards more imaging modalities, as demonstrated by Liebgott et al. [Lie+21], or even further application domains outside medical image segmentation, e.g. image to image translation, are possible future research topics.

The proposed domain generalization strategies in chapter 4.5, comprising the usage of Oktay et al.'s shape priors [OF+18], the infusion of contour information by edge enhancement, and the enforcement of shape aware feature learning by means of color augmentation, all result in improved segmentation results on an unseen CT target domain, although merely trained on a MR source domain. It is demonstrated, that the combination of all three strategies even outperforms the upper baseline, consisting of a U-Net, trained on the source domain and additionally fine-tuned on the target domain with additional target domain data. In the scope of this dissertation, only liver segmentation is considered. Therefore, for future research the extension to multi-organ segmentation should be considered. Furthermore, the MR source domain consists of both T1- and T2-weighted MR volumes, which could be additionally advantageous in the abstraction from intensity based features towards shape aware ones. For further research, the applicability of these methods for *single* source domain generalization towards unseen target domains should be investigated. Furthermore, the color maps have been manually selected, following a feature engineering approach for filter banks. The abstraction towards automated color map selection or even *learned* color maps, similar to learned features in CNNs, is also an open task to

be addressed in the future.

In chapter 5.1, topographical priors are incorporated by means of multitask learning for the extraction of small structures, such as necrotic tissue in large MR volumes. The results show major individual improvements regarding Hausdorff distance and in overall DSC. The DSC, however, shows large variation for both U-Net and the proposed method. This is due to the large effect a falsely classified pixel has on the overall DSC in case of *small* structures. Therefore, the question arises, whether the DSC is a suitable metric for segmentation tasks of small structures. This is also reflected by only considering the foreground dice loss during training (for both architectures), instead of using the mean of fore- and background loss.

With the ablations insights from chapter 4.4, a necrosis projection pipeline from 3D MR data to 2D x-ray images is proposed in chapter 6, combining the developed shape aware IE_2D-Net from chapter 4.3 and the topography aware multitask learning architecture from chapter 5.1 with a landmark based evolutionary registration strategy. Although the evaluation shows qualitative and quantitative good overall results, the used DSC score needs to be assessed with caution, since large differences in the field of view of the moving and fixed image may lead to aggravated DSC scores, although the visual alignment is satisfactory.
For future research, further combinations of shape and topography aware methods may be of interest to extract structures, that are restricted to a specific topographical location, which possesses inter-patient shape consistency. One example would be the extraction of liver lesions, which are restricted to the liver area, which again is consistent in shape across both patients and even modalities. The application of the proposed domain generalization approach from chapter 4.5 is a suitable strategy, if the images, in which liver and lesions are displayed, are of different modality.

Altogether, this last chapter and this thesis can be concluded by stating that although multiple new approaches and architecture designs regarding anatomical priors have been presented within this dissertation, Alan Turing's quote from the 1950's [Tur09] still holds in present time:

"*We can only see a short distance ahead, but we can see plenty there that needs to be done.*"

List of Figures

List of Tables

Bibliography

[AEZ16] Ramzi Ben Ali, Ridha Ejbali, and Mourad Zaied. "Detection and classification of dental caries in x-ray images using deep neural networks". In: *International Conference on Software Engineering Advances (ICSEA)*. 2016, p. 236.

[Aum+14] Gerhard Aumüller et al. "Anatomie". In: Duale Reihe. Stuttgart: Georg Thieme Verlag, 2014. Chap. 4.2 Standardverfahren.

[BM92] Paul J Besl and Neil D McKay. "Method for registration of 3-D shapes". In: *Sensor fusion IV: control paradigms and data structures*. Vol. 1611. International Society for Optics and Photonics. 1992, pp. 586–606.

[BPT14] Hakan Bilen, Marco Pedersoli, and Tinne Tuytelaars. "Weakly Supervised Detection with Posterior Regularization". In: *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014.

[Bre62] Hans J Bremermann. "Optimization through evolution and recombination". In: *Self-organizing systems* 93 (1962), p. 106.

[Bro+13] Esther E Bron et al. "Image registration improves human knee cartilage T1 mapping with delayed gadolinium-enhanced MRI of cartilage (dGEMRIC)". In: *European radiology* 23.1 (2013), pp. 246–252.

[Bui+19] Toan Duc Bui et al. "Multi-task Learning for Neonatal Brain Segmentation Using 3D Dense-Unet with Dense Attention Guided by Geodesic Distance". In: *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*. Springer, 2019, pp. 243–251.

[ÇA+16] Ö. Çiçek, A. Abdulkadir, et al. "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016, pp. 424–432.

[Cas+19] F Casalegno et al. "Caries detection with near-infrared transillumination using deep learning". In: *Journal of dental research* 98.11 (2019), pp. 1227–1233.

[CEK18] Joonhyang Choi, Hyunjun Eun, and Changick Kim. "Boosting proximal dental caries detection via combination of variational methods and convolutional neural network". In: *Journal of Signal Processing Systems* 90.1 (2018), pp. 87–97.

[CET98] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. "Active appearance models". In: *European conference on computer vision*. Springer. 1998, pp. 484–498.

[Cha+14] Shekhar S Chandra et al. "Focused shape models for hip joint segmentation in 3D magnetic resonance images". In: *Medical image analysis* 18.3 (2014), pp. 567–578.

[Che+20] Liangliang Cheng et al. "3D Printed Personalized Guide Plate in the Femoral Head Core Decompression". In: *BioMed Research International* 2020 (2020).

[Chr+16] Patrick Ferdinand Christ et al. "Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016, pp. 415–423.

[Chu+15] Chengwen Chu et al. "FACTS: Fully automatic CT segmentation of a hip joint". In: *Annals of biomedical engineering* 43.5 (2015), pp. 1247–1259.

[CK11] Peter I Corke and Oussama Khatib. *Robotics, vision and control: fundamental algorithms in MATLAB*. Vol. 73. Springer, 2011.

[CM92] Yang Chen and Gérard Medioni. "Object modelling by registration of multiple range images". In: *Image and vision computing* 10.3 (1992), pp. 145–155.

[Con+21] Pierre-Henri Conze et al. "Abdominal multi-organ segmentation with cascaded convolutional and adversarial deep networks". In: *Artificial Intelligence in Medicine* 117 (2021), p. 102109.

[CSS03] Daniel Cremers, Nir Sochen, and Christoph Schnörr. "Towards recognition based variational segmentation using shape priors and dynamic labeling". In: *International Conference on Scale-Space Theories in Computer Vision*. Springer. 2003, pp. 388–400.

[CT92] Timothy F. Cootes and Christopher J. Taylor. "Active shape models - 'smart snakes'". In: *BMVC92*. Springer, 1992, pp. 266–275.

[CV01] Tony F Chan and Luminita A Vese. "Active contours without edges". In: *IEEE Transactions on image processing* 10.2 (2001), pp. 266–277.

[CV+13] Kenneth Clark, Bruce Vendt, et al. "The Cancer Imaging Archive (TCIA): maintaining and operating a public information repository". In: *Journal of digital imaging* 26.6 (2013), pp. 1045–1057.

[DC15] Soma Datta and Nabendu Chaki. "Detection of dental caries lesion at early stage based on image analysis technique". In: *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*. IEEE, 2015.

[Den+09a] J. Deng et al. "ImageNet: A Large-Scale Hierarchical Image Database". In: *CVPR09*. 2009.

[Den+09b] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[DLY19] Shusil Dangi, Cristian A Linte, and Ziv Yaniv. "A distance map regularized CNN for cardiac cine MR image segmentation". In: *Medical physics* 46.12 (2019), pp. 5637–5651.

[DMJRM00] Roy De Maesschalck, Delphine Jouan-Rimbaud, and Désiré L Massart. "The mahalanobis distance". In: *Chemometrics and intelligent laboratory systems* 50.1 (2000), pp. 1–18.

[DX18] Nanqing Dong and Eric P Xing. "Few-shot semantic segmentation with prototype learning". In: *BMVC*. Vol. 3. 2018, p. 4.

[ES+03] Agoston E Eiben, James E Smith, et al. *Introduction to evolutionary computing*. Vol. 53. Springer, 2003.

[Eve+07] Mark Everingham et al. *The Pascal Visual Object Classes Challenge 2007*. `http://host.robots.ox.ac.uk/pascal/VOC/voc2007/index.html`. Accessed: 2022-04-23. 2007.

[Eve+10a] Mark Everingham et al. *The Pascal Visual Object Classes Challenge 2010*. `http://host.robots.ox.ac.uk/pascal/VOC/voc2010/index.html`. Accessed: 2022-04-23. 2010.

[Eve+10b] Mark Everingham et al. "The pascal visual object classes (voc) challenge". In: *International journal of computer vision* 88.2 (2010), pp. 303–338.

[Eve+15] Mark Everingham et al. "The pascal visual object classes challenge: A retrospective". In: *International journal of computer vision* 111.1 (2015), pp. 98–136.

[FB81] Martin A Fischler and Robert C Bolles. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In: *Communications of the ACM* 24.6 (1981), pp. 381–395.

[Fis+20] Maximilian Fischer et al. "A robust method for automatic identification of femoral landmarks, axes, planes and bone coordinate systems using surface models". In: *Scientific reports* 10.1 (2020), pp. 1–11.

[Fog98] David B. Fogel. "Artificial Intelligence through Simulated Evolution". In: *Evolutionary Computation: The Fossil Record*. 1998, pp. 227–296.

[FOW66] L.J. Fogel, A.J. Owens, and M.J. Walsh. *Artificial intelligence through simulated evolution*. Chichester, WS, UK: Wiley, 1966.

[GA14] Mina Gharenazifam and Ehsan Arbabi. "Anatomy-based 3D skeleton extraction from femur model". In: *Journal of Medical Engineering & Technology* 38.8 (2014), pp. 402–410.

[Gei+18] Robert Geirhos et al. "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In: *arXiv preprint arXiv:1811.12231* (2018).

[GF+16] R. Girdhar, D. F. Fouhey, et al. "Learning a Predictable and Generative Vector Representation for Objects". In: *European Conference on Computer Vision*. Springer. 2016, pp. 484–499.

[GG+18] Eli Gibson, Francesco Giganti, et al. "Automatic multi-organ segmentation on abdominal CT with dense v-networks". In: *IEEE transactions on medical imaging* 37.8 (2018), pp. 1822–1834.

[Gha+17] Mohsen Ghafoorian et al. "Transfer learning for domain adaptation in mri: Application in brain lesion segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2017, pp. 516–524.

[Gir+14] Ross Girshick et al. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[Gir15] Ross Girshick. "Fast R-CNN". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1440–1448.

[Gra06] L. Grady. "Random Walks for Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.11 (2006), pp. 1768–1783.

[Has] Muneeb ul Hassan. *Vgg16 – convolutional network for classification and detection*. `https://neurohive.io/en/popular-networks/vgg16/`. Accessed: 2021-12-22.

[Hat+19] Ali Hatamizadeh et al. "Deep Active Lesion Segmentation". In: *Machine Learning in Medical Imaging*. Ed. by Heung-Il Suk et al. Cham: Springer International Publishing, 2019, pp. 98–105. ISBN: 978-3-030-32692-0.

[He+16] K. He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[Hes+20] Linde S Hesse et al. "Intensity augmentation to improve generalizability of breast segmentation across different MRI scan protocols". In: *IEEE Transactions on Biomedical Engineering* 68.3 (2020), pp. 759–770.

[Hol73] John H Holland. "Genetic algorithms and the optimal allocation of trials". In: *SIAM journal on computing* 2.2 (1973), pp. 88–105.

[Hol92] John H Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[Hou62] Paul VC Hough. *Method and means for recognizing complex patterns*. US Patent 3,069,654. 1962.

[Hum+19] R. Hummel et al. "Caries Progression Rates Revisited: A Systematic Review". In: *Journal of Dental Research* 98 (July 2019), pp. 746–754.

[Hun07] J. D. Hunter. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95.

[IK+18] Fabian Isensee, Philipp Kickingereder, et al. "No new-net". In: *International MICCAI Brainlesion Workshop*. Springer. 2018, pp. 234–244.

[Ise+21] Fabian Isensee et al. "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation". In: *Nature methods* 18.2 (2021), pp. 203–211.

[Iso+17] Phillip Isola et al. "Image-to-Image Translation with Conditional Adversarial Networks". In: *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. 2017.

[Jia+18] Jue Jiang et al. "Tumor-aware, adversarial domain adaptation from ct to mri for lung cancer segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 777–785.

[JJ08] Ho-Ryong Jung and Moon-Ryul Jung. "An implicit active contour model for feature regions and lines". In: *Advances in Multimedia Modeling* (2008), pp. 35–44.

[Kai+09] Dagmar Kainmueller et al. "An articulated statistical shape model for accurate hip joint segmentation". In: *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*. IEEE. 2009, pp. 6345–6351.

[Kam+17] Konstantinos Kamnitsas et al. "Unsupervised domain adaptation in brain lesion segmentation with adversarial networks". In: *International conference on information processing in medical imaging*. Springer. 2017, pp. 597–609.

[Kav+21] A Emre Kavur et al. "CHAOS challenge-combined (CT-MR) healthy abdominal organ segmentation". In: *Medical Image Analysis* 69 (2021), p. 101950.

[KC+18] S. Kumar, S. Conjeti, et al. "InfiNet: Fully Convolutional Networks for Infant Brain MRI Segmentation". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE. 2018, pp. 145–148.

[Kho07] Kourosh Khoshelham. "Extending generalized hough transform to detect 3d objects in laser range data". In: *ISPRS Workshop on Laser Scanning and SilviLaser 2007, 12-14 September 2007, Espoo, Finland*. International Society for Photogrammetry and Remote Sensing. 2007.

[Kos+06] Suwadee Kositbowornchai et al. "An Artificial Neural Network for Detection of Simulated Dental Caries". en. In: *International Journal of Computer Assisted Radiology and Surgery* 1.2 (Aug. 2006), pp. 91–96. (Visited on 10/13/2020).

[Koz94] John R Koza. "Genetic programming as a means for programming computers by natural selection". In: *Statistics and computing* 4.2 (1994), pp. 87–112.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012, pp. 1097–1105. URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[KSH19] Christina Karam, Kenjiro Sugimoto, and Keigo Hirakawa. "Fast Convolutional Distance Transform". In: *IEEE Signal Processing Letters* 26.6 (2019), pp. 853–857.

[KWT88] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. "Snakes: Active contour models". In: *INTERNATIONAL JOURNAL OF COMPUTER VISION* 1.4 (1988), pp. 321–331.

[KZS15] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. "Siamese neural networks for one-shot image recognition". In: *ICML Deep Learning Workshop*. Vol. 2. 2015.

[Lam+19] Z. Lambert et al. "SegTHOR: Segmentation of Thoracic Organs at Risk in CT images". In: *arXiv preprint arXiv:1912.05950* (2019).

[LeC+89a] Yann LeCun et al. "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4 (1989), pp. 541–551.

[LeC+89b] Yann LeCun et al. "Handwritten digit recognition with a back propagation network". In: *Advances in neural information processing systems* 2 (1989).

[Li+16] Dong Li et al. “Weakly Supervised Object Localization With Progressive Domain Adaptation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Li+18] Bo Li et al. “Clinical value of 3D printing guide plate in core decompression plus porous bioceramics rod placement for the treatment of early osteonecrosis of the femoral head”. In: *Journal of orthopaedic surgery and research* 13.1 (2018), pp. 1–7.

[Lia+20] Yuan Liang et al. “OralCam: Enabling Self-Examination and Awareness of Oral Health Using a Smartphone Camera”. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–13.

[Lie+21] Annika Liebgott et al. “Automated Multi-Organ Segmentation in Pet Images Using Cascaded Training of a 3d U-Net and Convolutional Autoencoder”. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 1145–1149.

[Lin+12] Claudia Lindner et al. “Accurate fully automatic femur segmentation in pelvic radiographs using regression voting”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2012, pp. 353–360.

[Lin+14] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

[Liu+19] Lizheng Liu et al. “A Smart Dental Health-IoT Platform Based on Intelligent Hardware, Deep Learning, and Mobile Terminal”. In: *IEEE journal of biomedical and health informatics* 24.3 (2019), pp. 898–906.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[LX+15] BA Landman, Z Xu, et al. *MICCAI multi-atlas labeling beyond the cranial vault - workshop and challenge*. `https://doi.org/10.7303/syn3193805`. 2015.

[Ma+20] Jun Ma et al. “How Distance Transform Maps Boost Segmentation CNNs: An Empirical Study”. In: *Medical Imaging with Deep Learning*. 2020.

[May+16] Jonathan Mayes et al. “How smartphone technology is changing healthcare in developing countries”. In: *The Journal of Global Health at Columbia University* 6.2 (2016), pp. 36–38.

[MBE18] Claudio Michaelis, Matthias Bethge, and Alexander Ecker. “One-Shot Segmentation in Clutter”. In: *International Conference on Machine Learning*. 2018, pp. 3546–3555.

[MNA16] F. Milletari, N. Navab, and S.-A. Ahmadi. “V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation”. In: *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE. 2016, pp. 565–571.

[MP43] Warren S McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[MP69] Marvin Minsky and Seymour Papert. "Perceptrons.". In: (1969).

[Nav+19] Fernando Navarro et al. "Shape-aware complementary-task learning for multi-organ segmentation". In: *International Workshop on Machine Learning in Medical Imaging*. Springer. 2019, pp. 620–627.

[OF+18] O. Oktay, E. Ferrante, et al. "Anatomically Constrained Neural Networks (ACNNs): Application to Cardiac Image Enhancement and Segmentation". In: *IEEE Transactions on Medical Imaging* 37.2 (2018), pp. 384–395.

[Oqu+15] Maxime Oquab et al. "Is object localization for free?-weakly-supervised learning with convolutional neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 685–694.

[OS88] Stanley Osher and James A Sethian. "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations". In: *Journal of Computational Physics* 79.1 (1988), pp. 12 –49.

[Ots79] Nobuyuki Otsu. "A threshold selection method from gray-level histograms". In: *IEEE transactions on systems, man, and cybernetics* 9.1 (1979), pp. 62–66.

[PDP20] Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "Liver segmentation in ct with mri data: Zero-shot domain adaptation by contour extraction and shape priors". In: *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*. IEEE. 2020, pp. 1538–1542.

[PDP21a] Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "A Differentiable Convolutional Distance Transform Layer for Improved Image Segmentation". In: *Pattern Recognition: 42nd DAGM German Conference, DAGM GCPR 2020, Tübingen, Germany, September 28–October 1, 2020, Proceedings 42*. Springer. 2021, pp. 432–444.

[PDP21b] Duc Duy Pham, Gurbandurdy Dovletov, and Josef Pauli. "Using Anatomical Priors for Deep 3D One-shot Segmentation.". In: *BIOIMAGING*. 2021, pp. 174–181.

[Pha+18] Duc Duy Pham et al. "Polar appearance models: a fully automatic approach for femoral model initialization in MRI". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE. 2018, pp. 1002–1005.

[Pha+19a] Duc Duy Pham et al. "Deep learning with anatomical priors: imitating enhanced autoencoders in latent space for improved pelvic bone segmentation in MRI". In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE. 2019, pp. 1166–1169.

[Pha+19b] Duc Duy Pham et al. "Deep Segmentation Refinement with Result-Dependent Learning". In: *Bildverarbeitung für die Medizin 2019*. Wiesbaden: Springer Fachmedien Wiesbaden, 2019, pp. 49–54. ISBN: 978-3-658-25326-4.

[Pha+19c] Duc Duy Pham et al. “Gradient-Based Expanding Spherical Appearance Models for Femoral Model Initialization in MRI”. In: *Bildverarbeitung für die Medizin 2019*. Springer, 2019, pp. 43–48.

[Pha+20] Duc Duy Pham et al. “Multitask-Learning for the Extraction of Avascular Necrosis of the Femoral Head in MRI”. In: *Bildverarbeitung für die Medizin 2020*. Springer, 2020, pp. 150–155.

[Pha+21] Duc Duy Pham et al. “Fully vs. Weakly Supervised Caries Localization in Smartphone Images with CNNs”. In: *Pattern Recognition. ICPR International Workshops and Challenges*. Springer International Publishing, 2021, pp. 321–336.

[PK15] Cong-Bo Phan and Seungbum Koo. “Predicting anatomical landmarks and bone morphology of the femur using local region matching”. In: *International journal of computer assisted radiology and surgery* 10.11 (2015), pp. 1711–1719.

[PMGR01] N. Paragios, O. Mellina-Gottardo, and V. Ramesh. “Gradient vector flow fast geodesic active contours”. In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. Vol. 1. 2001, 67–73 vol.1.

[PRR02] Nikos Paragios, Mikael Rousson, and Visvanathan Ramesh. “Matching distance functions: A shape-to-area variational approach for global-to-local registration”. In: *European Conference on Computer Vision*. Springer. 2002, pp. 775–789.

[Rec65] Ingo Rechenberg. “Cybernetic solution path of an experimental problem”. In: *Royal Aircraft Establishment Library Translation 1122* (1965).

[Rec73] Ingo Rechenberg. *Evolutionsstrategie—Optimierung technischer Systeme nach Prinzipien der biologischen Information*. 1973.

[Rec78] Ingo Rechenberg. “Evolutionsstrategien”. In: *Simulationsmethoden in der Medizin und Biologie*. Springer, 1978, pp. 83–114.

[Ren+15] Shaoqing Ren et al. “Faster R-CNN: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems*. 2015, pp. 91–99.

[RF+16] Holger R. Roth, Amal Farag, et al. *Data From Pancreas-CT*. The Cancer Imaging Archive. `http://doi.org/10.7937/K9/TCIA.2016.tNB1kqBU`. 2016.

[RFB15] O. Ronneberger, P. Fischer, and T. Brox. “U-net: Convolutional Networks for Biomedical Image Segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2015, pp. 234–241.

[RHW86] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back propagating errors”. In: *nature* 323.6088 (1986), pp. 533–536.

[RL+15] Holger R Roth, Le Lu, et al. “Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation”. In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2015, pp. 556–564.

[Ros57] Frank Rosenblatt. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory, 1957.

[Ros58] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain.". In: *Psychological review* 65.6 (1958), p. 386.

[Ros61] Frank Rosenblatt. *Principles of neurodynamics. perceptrons and the theory of brain mechanisms*. Tech. rep. Cornell Aeronautical Lab Inc Buffalo NY, 1961.

[RP02] Mikael Rousson and Nikos Paragios. "Shape priors for level set representations". In: *European Conference on Computer Vision*. Springer. 2002, pp. 78–92.

[RV+17] H. Ravishankar, R. Venkataramani, et al. "Learning and Incorporating Shape Models for Semantic Segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2017, pp. 203–211.

[SB+16] Adam Santoro, Sergey Bartunov, et al. "One-shot learning with memory augmented neural networks". In: *arXiv preprint arXiv:1605.06065* (2016).

[Sch77] Hans-Paul Schwefel. "Evolutionsstrategien für die numerische optimierung". In: *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*. Springer, 1977, pp. 123–176.

[Sch93] Hans-Paul Paul Schwefel. *Evolution and optimum seeking: the sixth generation*. John Wiley & Sons, Inc., 1993.

[Sel+17] Ramprasaath R Selvaraju et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 618–626.

[SK+19] Alper Selver, Ali Emre Kavur, et al. *CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation*. `https://chaos.grand-challenge.org/Data/`. 2019.

[SM83] Gerard Salton and Michael J McGill. *Introduction to modern information retrieval*. mcgraw-hill, 1983.

[SMT08] Jérôme Schmid and Nadia Magnenat-Thalmann. "MRI bone segmentation using deformable models and shape priors". In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2008, pp. 119–126.

[Son+16] Bi Song et al. "Anatomy-Guided Brain Tumor Segmentation and Classification". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Springer International Publishing, 2016, pp. 162–170. ISBN: 978-3-319-55524-9.

[SRG14] T.P. Saravanan, M. Raj, and Kannaki Gopalakrishnan. "Identification of early caries in human tooth using histogram and power spectral analysis". In: *Middle - East Journal of Scientific Research* 20 (Jan. 2014), pp. 871–875.

[Sri+20] Adepu Srilatha et al. "Advanced diagnostic aids in dental caries – A review". In: *Journal of Global Oral Health* 2 (Feb. 2020), pp. 118–127.

[SSZ17] Jake Snell, Kevin Swersky, and Richard Zemel. "Prototypical networks for few-shot learning". In: *Advances in Neural Information Processing Systems*. 2017, pp. 4077–4087.

[Stu08] Student. "The probable error of a mean". In: *Biometrika* (1908), pp. 1–25.

[SZ15a] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1409.1556.

[SZ15b] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*. 2015.

[Tan+17] Min Tang et al. "A Deep Level Set Method for Image Segmentation". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings*. Springer. 2017, p. 126.

[Teu12] Christof Teuscher. *Foreword: Special issue on Alan Turing*. 2012.

[Tur09] Alan M Turing. "Computing machinery and intelligence". In: *Parsing the turing test*. Springer, 2009, pp. 23–65.

[Tur48] Alan Mathison Turing. *Intelligent machinery*. 1948.

[Ul] User:Scuba-limp. *Gesundes Hüftgelenk*. https://commons.wikimedia.org/wiki/File:Hueftgelenk-gesund.jpg. Accessed: 2022-04-21, Distributed under a CC-BY-SA license.

[VB+16] Oriol Vinyals, Charles Blundell, et al. "Matching networks for one shot learning". In: *Advances in neural information processing systems*. 2016, pp. 3630–3638.

[Ver+09] Tom Vercauteren et al. "Diffeomorphic demons: Efficient non-parametric image registration". In: *NeuroImage* 45.1 (2009), S61–S72.

[Vor+18] E. Vorontsov et al. "Liver lesion segmentation informed by joint liver segmentation". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. 2018, pp. 1332–1335.

[Wan+17] Wei Wang et al. "Patient-specific core decompression surgery for early-stage ischemic necrosis of the femoral head". In: *PloS one* 12.5 (2017), e0175366.

[Wan+19] Fang Wan et al. "C-mil: Continuation multiple instance learning for weakly supervised object detection". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 2199–2208.

[Wil92] Frank Wilcoxon. "Individual comparisons by ranking methods". In: *Breakthroughs in statistics*. Springer, 1992, pp. 196–202.

[Xia+13] Ying Xia et al. "Automated bone segmentation from large field of view 3D MR images of the hip joint". In: *Physics in medicine and biology* 58.20 (2013), p. 7375.

[XL+16] Zhoubing Xu, Christopher P Lee, et al. "Evaluation of six registration methods for the human abdomen on clinically acquired CT". In: *IEEE Transactions on Biomedical Engineering* 63.8 (2016), pp. 1563–1572.

[XP97] Chenyang Xu and Jerry L. Prince. "Gradient vector flow: A new external force for snakes". In: *In Proceedings of the Conference on Computer Vision and Pattern Recognition*. 1997, pp. 66–71.

[YNS14] Lassad Ben Younes, Yoshikazu Nakajima, and Toki Saito. "Fully automatic segmentation of the Femur from 3D-CT images using primitive shape recognition and statistical shape models". In: *International journal of computer assisted radiology and surgery* 9.2 (2014), pp. 189–196.

[Yok+09] Futoshi Yokota et al. "Automated segmentation of the femur and pelvis from 3D CT data of diseased hip using hierarchical statistical shape model of joint structure". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009* (2009), pp. 811–818.

[Zha+20a] Ling Zhang et al. "Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation". In: *IEEE transactions on medical imaging* 39.7 (2020), pp. 2531–2540.

[Zha+20b] Yipeng Zhang et al. "A Smartphone-Based System for Real-Time Early Childhood Caries Diagnosis". In: *Medical Ultrasound, and Preterm, Perinatal and Paediatric Image Analysis*. Springer, 2020, pp. 233–242.

[Zho+16] Bolei Zhou et al. "Learning deep features for discriminative localization". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2921–2929.

[Zhu+17] Jun-Yan Zhu et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networkss". In: *Computer Vision (ICCV), 2017 IEEE International Conference on*. 2017.

[Zor+01] Reza A Zoroofi et al. "Segmentation of avascular necrosis of the femoral head using 3-D MR images". In: *Computerized medical imaging and graphics* 25.6 (2001), pp. 511–521.

List of Abbreviations

CT	**C**omputer **T**omography
MR	**M**agnetic **R**esonance
MRI	**M**agnetic **R**esonance **I**maging
X-ray	**X-ray** image
MP	**M**cCulloch **P**itts
RP	**R**osenblatt **P**erceptron
MLP	**M**ulti **L**ayer **P**erceptron
MLP	**M**ulti **L**ayer **P**erceptron
FC	**F**ully **C**onnected
CNN	**C**onvolutional **N**eural **N**etwork
FCN	**F**ully **C**onvolutional **N**etwork
TPs	**T**rue **P**ositives
FPs	**F**alse **P**ositives
TNs	**T**rue **N**egatives
FNs	**F**alse **N**egatives
DSC	**D**ice **S**imilarity **C**oefficient
HD	**H**ausdorff **D**istance
sHD	**s**ymmetric **H**ausdorff **D**istance
PAMs	**P**olar **A**ppearance **M**odels
ICP	**I**terative **C**losest **P**oint
PCA	**P**rincipal **C**omponent **A**nalysis
FOV	**F**ield **O**f **V**iew
GESAMs	**G**radient based **E**xpanding **S**pherical **A**ppearance **M**odels
RANSAC	**Ran**dom **Sa**mpling **C**onsensus
LS	**L**evel **S**et
FSPR	**F**easible **S**phere **P**roposal **R**ate
GVF	**G**radient **V**ector **F**low
GAN	**G**enerative **A**dversarial **N**etwork
CDT	**C**onvolutional **D**istance **T**ransform
CAE	**C**onvolutional **A**uto**E**ncoder
IED	**I**mitating **E**ncoder - **D**ecoder
IE$_2$D	**I**mitating **E**ncoder - **E**nhanced **D**ecoder
IRE$_3$D	**I**mitating and **R**egularizing **E**ncoder$\underline{s}$ - **E**nhanced **D**ecoder
ACNN	**A**natomically **C**onstrained **N**eural **N**etwork
CHAOS	**C**ombined **H**ealthy **A**bdominal **O**rgan **S**egmentation
RAVD	**R**elative **A**bsolute **V**olume **D**ifference
ISDUE	**I**ntelligent **S**ystems group of the University of **Du**isburg-**E**ssen
TCIA	**T**he **C**ancer **I**maging **A**rchive
BTCV	**B**eyond **T**he **C**ranial **V**ault

pCT	**p**seudo **CT**
AVNFH	**A**vascular **N**ecrosis of the **F**emoral **H**ead
TEP	**T**otal **E**ndo-**P**rosthesis
ROI	**R**egion **O**f **I**nterest
MTL-Net	**M**ulti-**T**ask **L**earning Net
FIA	**F**ully **I**nformative **A**nnotation
ILA	**I**mage-**L**evel **A**nnotation
MRA	**M**outh **R**egion **A**nnotation
MIL	**M**ultiple **I**nstance **L**earning
R-CNN	**R**egion based **C**onvolutional **N**eural **N**etwork
RPN	**R**egion **P**roposal **N**etwork
CAM	**C**lass **A**ctivation **M**apping
GAP	**G**lobal **A**verage **P**ooling
Grad-CAM	**G**radient-weighted **C**lass **A**ctivation **M**apping
WSCDM	**W**eakly **S**upervised **C**aries **D**etection **M**ethod
WSCDM-LC	**W**eakly **S**upervised **C**aries **D**etection **M**ethod with **L**ocation **C**onstraints
mAP	**m**ean **A**verage **P**recision
IoU	**I**ntersection **o**ver **U**nion
AP	**A**verage **P**recision
AUC	**AU**nder the **C**urve
VOC	**V**isual **O**bject **C**lasses
COCO	**C**ommon **O**bjects in **Co**ntext

List of Symbols

Ω	arbitrary index space of an image or a volume
Ω_{2D}	index space of a 2D image
Ω_{3D}	index space of a 3D image
$\mathbb{R}$	real number set
$\mathbb{R}^n$	n-dimensional real number set
$I : \Omega \to \mathbb{R}$	image representation as a mapping
$GT : \Omega \to \{0, 1\}^C$	ground truth representation as a mapping
C	number of classes
$\mathscr{C}$	set of all possible classes
I_m	moving image
I_f	fixed image
GT_m	ground truth of moving image
GT_f	ground truth of fixed image
$\mathscr{I}$	set of all images
$\mathscr{I}_{2D}$	set of all 2D images
$\mathscr{I}_{3D}$	set of all 3D images
$T : \mathscr{I} \to \mathscr{I}$	image transform
Θ	set of parameters
$\mathscr{L}$	loss function
$\Gamma : [0, 1] \to \mathbb{R}^n$	parameterized curve in n-dimensional space
E	energy functional
$\Phi : \Omega \to \mathbb{R}$	embedding function
∇	gradient operator
$\mathbb{N}\backslash\{0\}$	natural number set without zero
ζ	threshold
f_p	propagation function
f_a	activation function
x	arbitrary incoming signal
$w^t_{h,i,j}$	weight between layers h and $h+1$ from i-th neuron to j-th neuron at time t
κ	kernel for convolution
$*$	convolutional operator
$\odot$	cross-correlation operator
n_{bins}	number of intensity bins
$\mathscr{B}_0, \ldots, \mathscr{B}_{n_{bins}-1}$	intensity bins
${}_kI$	k-th slice from image stack
${}_kGT$	k-th slice from ground truth stack

$\chi_k(p,j)$	indicator function for k-th slice if intensity ${}_kI(p)$ is in j-th intensity bin
h_k	normalized intensity histogram for k-th slice
v_k	feature vector of k-th slice
$\overline{v}$	mean feature vector
n_{PCA}	number of eigenvectors
$\mathscr{U} := (u_0, \ldots, u_{n_{PCA}-1})$	eigenvectors
$\mathscr{E} := \{\lambda_0, \ldots, \lambda_{n_{PCA}-1}\}$	eigenvalues
$\mathscr{M} := (\mathscr{U}, \mathscr{E}, \overline{v})$	first part of the Polar Appearance Models
n_{PC}	intensity profile length in polar space
$\tilde{K}$	number of intensity profiles in polar space
$\tilde{v}_0, \ldots, \tilde{v}_{\tilde{K}-1}$	intensity profiles in polar space
$\overline{\tilde{v}}$	mean intensity profile
$\tilde{n}_{PCA}$	number of eigenvectors
$\tilde{\mathscr{U}} := (\tilde{u}_0, \ldots, \tilde{u}_{\tilde{n}_{PCA}-1})$	eigenvectors
$\tilde{\mathscr{E}} := \{\tilde{\lambda}_0, \ldots, \tilde{\lambda}_{\tilde{n}_{PCA}-1}\}$	eigenvalues
$\tilde{\mathscr{M}} := (\tilde{\mathscr{U}}, \tilde{\mathscr{E}}, \overline{\tilde{v}})$	second part of the Polar Appearance Models
$\mathscr{P} := (\mathscr{M}, \tilde{\mathscr{M}})$	complete Polar Appearance Models
$s(p)$	similarity of intensity distribution within Hough circle of center point p to learned distribution in $\mathscr{M}$
$\hat{M}$	total number of axial slices in unseen MR volume
P_i	set of detected candidate points from Hough Transform of i-th axial slice
$c(p_{i-1}, p_i)$	cost of connecting point p_{i-1} to p_i
$w_{(\cdot)}$	weight
$F_i(p_i)$	cost function of minimal costs for path until $p_i \in P_i$
$\mu(p_i)$	mean intensity within Hough circle of center point $p_i \in P_i$
$var(p_i)$	intensity variance within Hough circle of center point $p_i \in P_i$
$r(p_i)$	radius of Hough circle of center point $p_i \in P_i$
$\tilde{\mu}(p_{i-1})$	mean intensity within Hough circles of all (considered) center points until $p_{i-1} \in P_{i-1}$
$\overline{p_{i-1}}$	mean center point position of all (considered) Hough center points until $p_{i-1} \in P_{i-1}$
$\emptyset$	mean score
r_i	radius of fitted sphere in GT_i for GESAMs approach
bw_i	bandwidth of outer neighborhood region in GESAMs approach
$v_i^{in}, v_i^{out} \in [0,1]^{n_{bins}}$	intensity distribution feature vectors for the inner and outer region
$\mathscr{U}^{(\cdot)}$	matrix of eigenvectors from feature variant $(\cdot)$ for GESAMs
$\overline{v}^{(\cdot)}$	mean feature vector of variant $(\cdot)$ for GESAMs
$\mathscr{M}^{(\cdot)}$	spherical appearance model of feature variant $(\cdot)$ for GESAMs
r_{max}	maximal radius of the fitted spheres
r_{min}	minimal radius of the fitted spheres
σ_r	standard deviation of the detected spheres' radii
$\mathscr{G}$	GESAMs model
$\boldsymbol{\mathscr{g}}$	Gaussian kernel
$\lvert\nabla\{I * \boldsymbol{\mathscr{g}}\}\rvert$	smoothed gradient magnitude volume
$\mu_{\lvert\nabla\{I*\boldsymbol{\mathscr{g}}\}\rvert>0}$	mean of all positive values in the smoothed gradient magnitude volume
I_∇	binary volume of constrained sample domain by threshold

$n_{circles}$	number of considered circles for domain restriction by Hough Transform
$I_\circ$	binary volume containing most circle intersections from different 3D orientations
$I_{\nabla,\circ}$	Hadamard product of $I_\circ$ and $I_\circ$
$n_{sp} \geq 1$	minimal number of sufficient sphere proposals
$n_{nsp} \geq 1$	maximum number of subsequent non-sufficient proposals
p_{sc}	sphere candidate
$r(p_{sc})$	radius of sphere candidate p_{sc}
$v(p_{sc})^{(\cdot)}$	feature vector of variant $(\cdot)$ depending on sphere candidate p_{sc}
$s^{(\cdot)}(p_{sc})$	squared Mahalanobis distance to learned mean feature vector of variant $(\cdot)$ depending on sphere candidate p_{sc}
$n_{inlier}(p_{sc})$	number of sample points within a narrow band depending on sphere candidate p_{sc}
$\varepsilon > 0$	width of narrow band
$n_{inlier,scaled}(p_{sc})$	scaled number of sample points within a narrow band depending on sphere candidate p_{sc}
$\mu_{inliers}(p_{sc})$	mean of all inliers for a sphere candidate p_{sc}
$d_{inlier}(p_{sc})$	Euclidean distance of p_{sc} to $\mu_{inliers}(p_{sc})$
$S^{in}_{p_{sc}} : \Omega \to \{0,1\}$	indicator function of whether a point $p \in \Omega$ is within inner region of sphere candidate p_{sc}
$S^{out}_{p_{sc}} : \Omega \to \{0,1\}$	indicator function of whether a point $p \in \Omega$ is within outer boundary of sphere candidate p_{sc}
$\eta(p_{sc})$	homogeneity of sphere candidate p_{sc}
$\sigma^2(p_{sc})$	intensity variance in outer boundary of sphere candidate p_{sc}
$n_{dead}(p_{sc})$	number of dead edge points for a sphere candidate p_{sc}
P_{sc}	set of all proposed sphere candidate center points from the structured sampling step
$c(p_{sc})$	cost function of selecting sphere candidate p_{sc}
$E_{CV}(\Phi)$	energy functional by Chan and Vese depending on embedding function Φ
μ_{in}, μ_{out}	mean intensities of inner and outer contour region
$d(\cdot,\cdot)$	distance function
$D_I : \Omega \to \mathbb{R}^+_0$	distance transform of binary image I
$d_1, \ldots, d_n$	n scalar distances
$mathcal L_{(\cdot)}$	loss function for training CNNs
$\Theta_{(\cdot)}$	trainable weight configuration
f_{enc_p}	prior encoder
f_{enc_i}	imitating encoder
g_{dec}	generative decoder
h_{unet}	U-Net module
$\mathscr{I}_{gray}$	space of gray scale images
$\mathscr{I}_{color}$	space of color images
$CMAP : \mathscr{I}_{gray} \to \mathscr{I}_{color}$	color map
F_k	k-th feature map
$\alpha_k^{(c)}$	influence of k-th feature map on class assignment of class c
$A^{(c)}_{Grad-CAM}$	Grad-CAM for class c

R	rotation matrix
S	scaling matrix
$\vec{t}$	translation vector

DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

ub | universitäts bibliothek

Diese Dissertation wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt und liegt auch als Print-Version vor.

DOI: 10.17185/duepublico/78355
URN: urn:nbn:de:hbz:465-20230428-134430-0

Dieses Werk kann unter einer Creative Commons Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 Lizenz (CC BY-SA 4.0) genutzt werden.