AUTOMATIC LOCALIZATION OF SPATIALLY CORRELATED KEY POINTS IN MEDICAL IMAGES

ALEXANDER OLIVER MADER

A thesis submitted to the *Faculty of Engineering* at the *Kiel University* (*Christian-Albrechts-Universität zu Kiel*) in fulfillment of the requirements for the degree of *Doktoringenieur* (*Dr.-Ing.*).

Kiel, 2021

1. Gutachter: Prof. Dr. rer. nat. Carsten Meyer
Institut für Informatik
Christian-Albrechts-Universität zu Kiel

2. Gutachter: Prof. Dr.-Ing. Reinhard Koch
Institut für Informatik
Christian-Albrechts-Universität zu Kiel

Datum der mündlichen Prüfung: 29. März 2021

ABSTRACT

The task of object localization in medical images is a corner stone of automatic image processing and a prerequisite for other medical imaging tasks. In this thesis, we present a general framework for the automatic detection and localization of spatially correlated key points in medical images based on a conditional random field (CRF). The problem of selecting suitable potential functions (knowledge sources) and defining a reasonable graph topology w. r. t. the dataset is automated by our proposed data-driven CRF optimization.

We show how our fairly simple setup can be applied to different medical datasets involving different image dimensionalities (i. e., 2D and 3D), image modalities (i. e., X-ray, CT, MRI) and target objects ranging from 2 to 102 distinct key points by automatically adapting the CRF to the dataset. While the used general "default" configuration represents an easy to transfer setup, it already outperforms other state-of-the-art methods on three out of four datasets. By slightly gearing the proposed approach to the fourth dataset, we further illustrate that the approach is capable of reaching state-of-the-art performance of highly sophisticated and data-specific deep-learning-based approaches.

Additionally, we suggest and evaluate solutions for common problems of graph-based approaches such as the reduced search space and thus the potential exclusion of the correct solution, better handling of spatial outliers using latent variables and the incorporation of invariant higher order potential functions. Each extension is evaluated in detail and the whole method is additionally compared to a rivaling convolutional-neural-network-based approach on a hard problem (i. e., the localization of many locally similar repetitive target key points) in terms of exploiting the spatial correlation. Finally, we illustrate how follow-up tasks—segmentation in this case—may benefit from a correct localization by reaching state-of-the-art performance using off-the-shelve methods in combination with our proposed method.

ZUSAMMENFASSUNG

Objektlokalisation in medizinischen Bildern ist eine Kernaufgabe der automatischen Bildverarbeitung und Voraussetzung für andere Prozesse in der medizinischen Bilderverarbeitung. In dieser Thesis präsentieren wir einen generellen Ansatz zur automatischen Detektion und Lokalisation von räumlich korrelierten Schlüsselpunkten in medizinischen Bildern basierend auf einem Conditional Random Field (CRF). Die problematische Auswahl geeigneter Potentialfunktionen (Wissensquellen) ebenso wie die Definition einer geeigneten Graphtopologie in Bezug auf den Datensatz wird durch unsere vorgeschlagene datengetriebene Optimierung automatisiert.

Wir zeigen, wie unser vergleichsweise einfacher Aufbau auf verschiedene medizinische Datensätze mit unterschiedlichen Bilddimensionen (2D und 3D), Bildmodalitäten (Röntgen, CT und MRI) und Zielobjekten mit bis zu 102 Schlüsselpunkten angewandt werden kann, indem das CRF automatisch an den Datensatz angepasst wird. Obwohl die genutzte allgemeine „Standardkonfiguration“ einen einfach zu transferierenden Aufbau darstellt, übertrifft diese aber auf drei von vier Datensätzen bereits andere Methoden, die dem Stand der Technik entsprechen. Durch leichtes Anpassen der Methode an den vierten Datensatz können wir zeigen, dass die Methode die Erfolgsrate von hoch entwickelten und Datensatz-spezifischen Methoden auf Deep-Learning-Basis erreicht.

Weiterhin schlagen wir Lösungen für gängige Probleme von Graph-basierten Methoden vor und evaluieren sie im Detail. Dazu gehören die Reduktion des Suchraums und dem möglicherweise daraus resultierenden Ausschluss der korrekten Lösung, die bessere Handhabung von räumlichen Ausreißern mit Hilfe von latenten Variablen und die Aufnahme von invarianten Potentialfunktionen höherer Ordnung. Jede Erweiterung wird für sich analysiert und die gesamte Methode wird zusätzlich gegen einen alternativen Ansatz auf Basis von Faltungsnetzen auf einem anspruchsvollen Problem (die Lokalisation von sehr vielen lokal ähnlichen und repetitiven Zielpunkten) im Hinblick auf die Nutzung der räumlichen Korrelation hin untersucht. Abschließend zeigen wir, wie nachfolgende Arbeitsschritte, Objektsegmentierung in diesem Fall, von einer korrekten Lokalisation profitieren können, indem wir eine Standardlösung zur Segmentierung zielgenau anwenden und damit in der Lage sind, den Stand der Technik zu erreichen.

ACKNOWLEDGMENTS

Many different people contributed to this work in one way or another. Here, I would like to seize the opportunity to express my gratitude.

First of all I am very grateful to Prof. Carsten Meyer and Prof. Hauke Schramm from the Kiel University of Applied Sciences for raising my interest in image processing, pattern recognition and machine learning during my Master studies, and for introducing me to the field of medical image processing and the diverse difficulties therein.

I am deeply indebted to my mentor Prof. Carsten Meyer: He not only allowed and encouraged me to participate in his working group, but he also offered me the opportunity to write this thesis under his supervision. The countless inspiring—and sometimes controversial—discussions in combination with his motivation, experience, feedback and guidance helped me to cope with all the difficulties throughout the course of my PhD studies.

I am also very thankful to my industrial advisors Cristian Lorenz, Jens von Berg and Martin Bergtholdt from Philips Research Hamburg. Their great expertise and continuous support throughout this thesis were tremendously useful for my work.

Furthermore, I am very thankful to Prof. Jan Modersitzki from the University of Lübeck for his support from the beginning and offering me the chance to discuss my work in his working group. The thorough discussions with him in combination with his unique perspective on things were of immense help to me.

I would also like to thank my working group colleagues Eric Gabriel, Gordon Böer, Ferdinand Hahmann and Tanja Elß for not only making the time spent in the lab fun by providing help and countless enjoyable moments, but also outside the lab while playing some "SquareBall". Additionally, I would also acknowledge the many students whose projects and theses I supervised and who helped me to grow as a person. In particular, Prateek Jitendra Bhatt, Claudia Hoven, Jaro Richter, Malte Hecht, Eren Bora Yilmaz, Nhut Hai Huynh, David Kohn, Uchechukwu Okereke, Alexander Fabritz, Sayali Patkar, Oliver Fritsche and Thorsten Carlsson.

Finally, I am grateful to the Federal Ministry of Education and Research for financially supporting this work under the grant 03FH013IX5 and to Philips Research Hamburg for offering me a place to work and the accompanying benefits during my PhD studies.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2D	two **d**imensional	78
3D	three **d**imensional	78
AAM	**a**ctive **a**ppearance **m**odels	14
ANN	**a**rtificial **n**eural **n**etwork	25
AP	**a**nterop**o**sterior	81
ASM	**a**ctive **s**hape **m**odels	13
BRIEF	**b**inary **r**obust **i**ndependent **e**lementary **f**eatures	47
CNN	**c**onvolutional **n**eural **n**etwork	28
CPM	**c**onvolutional **p**ose **m**achine	70
CRF	**c**onditional **r**andom **f**ield	34
CT	**c**omputed **t**omography	20
FN	**f**alse **n**egative	96
FP	**f**alse **p**ositive	96
GHT	**g**eneralized **H**ough **t**ransform	9
GPGPU	**g**eneral-**p**urpose computing on **g**raphics **p**rocessing **u**nits	28
GPU	**g**raphics **p**rocessing **u**nit	28
HU	**H**ounsfield **u**nits	21
IVD	inter**v**ertebral **d**isc	91
L-TP	**l**ocalized **t**rue **p**ositive	97
LSTM	**l**ong **s**hort-**t**erm **m**emory	26
M-TP	**m**is-localized **t**rue **p**ositive	97
MAP	**m**aximum **a** **p**osteriori	32
MCMC	**M**arkov **c**hain **M**onte **C**arlo	68
MLE	**m**aximum **l**ikelihood **e**stimation	57
MPE	**m**ost **p**robable **e**xplanation	32
MRF	**M**arkov **r**andom **f**ield	30
MRI	**m**agnetic **r**esonance **i**maging	21
NMS	**n**on-**m**aximum **s**uppression	37
PA	**p**ostero**a**nterior	78
PCA	**p**rincipal **c**omponent **a**nalysis	13
PDM	**p**oint **d**istribution **m**odel	14
PET	**p**ositron **e**mission **t**omography	20

PGM	**p**robabilistic **g**raphical **m**odel	29
px	**p**i**x**el	78
ReLU	**r**ectified **l**inear **u**nit	28
RF	**r**andom **f**orest	25
RNN	**r**ecurrent **n**eural **n**etwork	26
RTE	**r**egression **t**ree **e**nsemble	47
SGD	**s**tochastic **g**radient **d**escent	53
SSE	**s**um of **s**quared **e**rrors	53
SVM	**s**upport **v**ector **m**achine	10
TN	**t**rue **n**egative	96
TP	**t**rue **p**ositive	96
vx	**v**o**x**el	78

LIST OF SYMBOLS

$\boldsymbol{A}_i$	size of the i-th target object	47
$\boldsymbol{B}$	potential function "missing" energies	40
β_f	"missing" value of the f-th potential function	40
c_f	clique associated with the f-th potential function	40
C_f	arity of the f-th potential function	40
D	image dimensionality	37
$\mathcal{D}$	images	66
$\mathcal{D}_{\text{graph}}$	graph-dedicated training images	104
$\mathcal{D}_{\text{pots}}$	potential-dedicated training images	104
$\mathcal{D}_{\text{test}}$	test images	104
$\mathcal{D}_{\text{train}}$	training images	104
$\mathcal{D}_{\text{val}}$	validation images	104
δ	training sample weighting	66
E	energy of the graphical model	40
e	error function	66
η	learning rate	68
F	number of potential functions	39
$\boldsymbol{I}$	image intensities	37
K	number of images	57
K_{test}	number of test images	95
K_{train}	number of training images	95
L	loss function	66
$\boldsymbol{\Lambda}$	potential function weights	40
λ_f	weight of the f-th potential function	40
m	energy margin	66
$\mathbb{N}$	natural numbers	37
N	number of key points	37
n_i	number of loc. hypotheses of the i-th key point	38
o_i^k	outcome type of the i-th key point in the k-th image	98
Ω	weight regularization	66
ϕ	potential function	39
Φ	pool of potential functions	39

ψ	negative log transformed (neg-log) potential function	40
Ψ	pool of neg-log potential functions	42
Ψ_i	neg-log potential functions depending on i-th key point	42
$\mathbb{R}$	real numbers	37
R	localization threshold	49
$\boldsymbol{S}$	random variables	39
S_i	random variable of the i-th key point	39
$\mathcal{S}$	joint value space of the distribution	38
$\mathcal{S}_i$	value space of the i-th random variable	38
$\boldsymbol{s}$	joint values of all random variables	38
$\boldsymbol{s}^+$	"correct" configuration	66
$\boldsymbol{s}^-$	best "incorrect" configuration	66
s_i	value of the i-th random variable	38
$\overset{\star}{s}_i$	correct value of the i-th random variable	96
$\hat{\boldsymbol{s}}$	selected values of all random variables	38
$\hat{s}_i$	selected value of the i-th random variable	42
T_i	number of decision trees used for the i-th target object	46
$\boldsymbol{T}$	affine transformation of the coordinate system	62
T^{s}	global isotropic scaling factor	63
θ	weight regularization factor	66
V_i	number of features used for the i-th target object	47
$\mathcal{X}_i$	localization hypotheses of the i-th key point	38
$\boldsymbol{x}_{i,j}$	j-th localization hypothesis of the i-th key point	38
$\overset{\star}{\boldsymbol{x}}_i$	annotated position of the i-th key point	43
$\hat{\boldsymbol{x}}_i$	predicted position of the i-th key point	96
$\tilde{\boldsymbol{x}}_i$	refined position of the i-th key point	42
$\boldsymbol{Y}_i$	heatmap of the i-th key point	37
Z	partition function	31

Part I

SETTING THE STAGE

1 INTRODUCTION

The task of object localization in digital images has a long history in computer science. Various real-world tasks, especially in the field of automation, require some sort of object localization, either as prerequisite for follow-up tasks or object localization in itself. Some notable fields are industrial robotic automation, natural scene analysis and medical imaging. However, the underlying conditions in each one are quite different.

For example, localizing screws in a box to get picked up by a stationary robot is a quite different task compared to localizing humans in arbitrary natural images. The first task is described by target objects showing very little variability as well as the environment which can be controlled very well. The second task has a lot more degrees of freedom due to the high object variability as well as the uncontrollable environmental setting. Hence, both tasks might require different approaches to achieve satisfactory results. The notion of satisfactory also varies quite drastically and puts emphasis on different parts of the method development.

1.1 MOTIVATION

The above considerations are even more true in medical imaging, where object detection and localization are corner stones of medical image processing. This ranges from simple tasks like deriving measurements (e. g., landmark distance estimation [183] or spinal Cobb angle estimation [39]) and deriving classifications (e. g., vertebra fracture classification [83]), over planning interventions [109] to augmenting the clinical workflow (e. g., automatic collimation calibration [176] or live instrument tracking [142]). Despite being useful in itself, it is also useful for other important tasks like image registration (i. e., establishing key point correspondences [214, 32]) or model-based segmentation (i. e., initial model placement [29]). Hence, the development of general object detection and localization methods can have a severe influence on medical imaging.

The definition of a satisfactory result w. r. t. these different tasks is also quite different to, e. g., natural scene analysis, and steers method development in potential distinct directions. In medical imaging, an emphasis is placed on robust and not overconfident methods, while real time operation is mostly—not always though—a lesser concern. It is also important to understand the limits of a method and to know

possible sources of errors. In terms of a clinical workflow, fully automatic approaches are desired, but the possibility to incorporate expert knowledge at test time in case of errors (semi-automatic operation) increases their practicality.

At the same time, the challenges being faced in medical imaging are quite different to, e. g., the ones found in natural scene analysis. Firstly, instead of operating on two-dimensional color images, multiple imaging modalities like X-ray, computed tomography (CT) and magnetic resonance imaging (MRI) are widely used. The different modalities lead to images with different dimensions ranging from simple two dimensional (2D) to temporal multi-channel three dimensional (3D) as well as different characteristics of pixel intensities. For example, CT provides (normalized) Hounsfield units, where as MRI provides unnormalized pixel intensities, which may additionally depend on the scanner type. Similarly to other imaging methods, these may also suffer from quality degradation, including intended one, e. g., when considering the tradeoff between radiation exposure and image quality. Secondly, the imaged anatomy can vary quite drastically. There are rigid structures like the brain, but also dynamic ones like the human body represented by connected deflectable body parts. Each of those structures also has some inherent variability, consider for example different patients and different ages. Furthermore, those anatomies might show pathological changes or contain artificial medical implants. Thirdly, the number of medical images is quite small in contrast to conventional imaging, especially annotated and publicly available images are rather scarce. This is even more true for pathological cases, which makes it hard to obtain datasets representing the underlying distribution and thus poses a severe challenge for machine-learning-based approaches.

Given this large set of combinations between task settings and challenges, few approaches try to tackle the problem of *general* object detection and localization. Instead, task-specific approaches are developed and tuned to the respective task and dataset at hand.

1.2 CONTRIBUTIONS

The main contribution of this thesis is the development and thorough evaluation of a general and transferable system for the automatic detection and localization of spatially correlated key points in digital medical images. The first task corresponds to the classification whether a labeled key point is contained in an image, while the second task corresponds to pinpointing the position of the detected key points. In this setting, the key points might be derived from anatomical objects or correspond to well-defined anatomical landmarks.

The proposed system utilizes a conditional random field (CRF) to model the spatial correlation between key points and uses different

knowledge sources to describe the likelihood of a certain key point constellation. The detection task is solved by introducing a dedicated "missing" label, which can be assigned to a key point instead of a position. Inference is performed in two steps. First, local appearance models are used to derive likely positions for each key point. Second, the combination of likely positions and "missing" labels is searched for the most likely constellation w. r. t. the confidence of the local appearance models and spatial non-unary regularization terms, both modeled as potential functions in the CRF.

Two methods to build local appearance models are suggested and evaluated. The first utilizes ensembles of regression trees to predict likelihoods for position-specific feature vectors. Their success in applied medical imaging w. r. t. high target performance and low training and test time in combination with a well-established foundation renders them ideal for such a task. The second method utilizes the well known U-Net convolutional neural network (CNN) architecture to—similarly—regress likelihoods while removing the heuristic feature selection. In combination with a graphics processing unit (GPU), CNNs often provide superior performance at a very low runtime. Both approaches build upon published and proven methods and have been adapted to fit into the proposed framework.

In order to exploit the spatial correlations and evaluate local constellations, binary and ternary potential functions are proposed that utilize simple spatial statistics. The spatial measures being used capture different spatial characteristics while being quick to compute and easy to estimate. For the binary potentials, the spatial measures distance, rotation and vector are used, each providing different invariance properties, i. e., rotation, scaling and non-invariant, respectively. Two novel ternary potentials are proposed, creating relative measures from distance and rotation in order to be both rotation and scaling invariant. Since this is also achievable with binary potentials utilizing latent variables, the ternary potentials are explicitly compared to such a setup.

A central problem of graph-based methods that hinders them being easily transferable is the problem of defining the graph topology and selecting suitable potential functions, since these have to be specifically optimized for each dataset and target structure. In order to solve these problems, an automatic data-driven learning scheme is proposed. It allows to quickly adapt the method to a dataset by weighting the different potential functions and estimating the "missing" label energies while optimizing the graph topology at the same time. The key idea is to start from a fully connected graph loaded with various potential functions taken from a pool of potential functions and optimize the weighting of each one. In addition to evaluating the importance of each potential function, the optimization also refines the graph topology by removing unnecessary (i. e., zero-weighted) potential functions. The

framework is geared towards not making any assumptions about the potential functions and is thus inclusive w. r. t. other potential functions proposed in the literature and profits from future research in this area, which also applies to the development of local appearance models. This allows to easily apply the method to a dataset with different target key points.

The easy transferability is empirically verified in great detail by applying the approach in a "default" configuration to various medical datasets of different imaging modality, image dimension, image resolution, imaged anatomy and the number of targeted key points. To put the results in perspective, they are compared to the state of the art on the respective datasets. We show how the "default" configuration is able to outperform non-deep-learning-based approaches without any adjustments on three out of four datasets. Furthermore, we illustrate how the approach can be easily geared towards a specific dataset in order to improve upon the initial results and reach the performance of highly task-specific deep-learning-based approaches on the fourth dataset as well.

Another common problem in graph-based (e. g., CRF) approaches is the reduction of the search space from the image domain to a set of likely positions in order to render inference in non-tree-shaped graphs feasible (consider the combinatorial complexity of, e. g., a high-resolution 3D image and many target key points), since this reduction might exclude the correct solution from the search space. To solve this problem, a novel "refine" label is introduced by changing the semantic meaning of the "missing" label to identify such cases using a first global inference, followed by efficient local optimizations of the "refine"-labeled key points over the whole image domain instead of just the set of likely positions on small subgraphs. We illustrate how this proposal is able to deliver robust and accurate results on the open problem of labeling posterior ribs in chest radiographs.

Following the recent trend of replacing older methods with CNN-based approaches, we additionally compare the chosen graph-based modelling of key point dependencies to a state-of-the-art CNN architecture that tries to implicitly capture the key point constellation on the unsolved problem of localizing a large amount of spinal key points in volumetric images. We adapt the CNN architecture to the target problem and suggest a simple refinement step to overcome the necessary downsampling (induced by the large memory demands of such methods in combination with a large amount of target key points) of the architecture and thus the reduced localization accuracy. It is shown that both approaches perform on par with each other but with subtle differences, which highlights different benefits of both approaches.

Finally, we evaluate the localization approach in terms of a follow-up task: the segmentation of small repetitive structures. It is illustrated how a correct localization of such small structures in combination with an off-the-shelve segmentation network is able to reach state-of-the-art performance (favorably) on the respective dataset.

1.3 PUBLICATIONS

Various aspects of the research described in this thesis have already been published in the following seven peer-reviewed papers:

- "Efficient Epiphyses Localization Using Regression Tree Ensembles and a Conditional Random Field" [136] by Alexander Oliver Mader, Hauke Schramm and Carsten Meyer. This paper was presented at the German conference *Bildverarbeitung für die Medizin* (*BVM*) in 2017.
- "Detection and Localization of Landmarks in the Lower Extremities Using an Automatically Learned Conditional Random Field" [133] by Alexander Oliver Mader, Cristian Lorenz, Martin Bergtholdt, Jens von Berg, Hauke Schramm, Jan Modersitzki and Carsten Meyer. This paper was presented at the workshop *Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics* (*GRAIL*), which was part of the *20th International Conference on Medical Image Computing and Computer Assisted Intervention* (*MICCAI*) in 2017. It was honored with the *GRAIL Best Paper Award*.
- "Localization and Labeling of Posterior Ribs in Chest Radiographs Using a CRF-regularized FCN with Local Refinement" [130] by Alexander Oliver Mader, Jens von Berg, Alexander Fabritz, Cristian Lorenz and Carsten Meyer. This paper was presented at the *21st International Conference on Medical Image Computing and Computer Assisted Intervention* (*MICCAI*) in 2018.
- "A Novel Approach to Handle Inference in Discrete Markov Networks with Large Label Sets" [131] by Alexander Oliver Mader, Jens von Berg, Cristian Lorenz and Carsten Meyer. This paper was presented at the *9th International Conference on Probabilistic Graphical Models* (*PGM*) in 2018.
- "Detection and localization of spatially correlated point landmarks in medical images using an automatically learned conditional random field" [134] by Alexander Oliver Mader, Cristian Lorenz, Martin Bergtholdt, Jens von Berg, Hauke Schramm, Jan Modersitzki and Carsten Meyer. This paper was published in the Elsevier journal *Computer Vision and Image Understanding* (*CVIU*) in 2018.

- "A General Framework for Localizing and Locally Segmenting Correlated Objects: A Case Study on Intervertebral Discs in Multi-Modality MR Images" [135] by Alexander Oliver Mader, Cristian Lorenz, and Carsten Meyer. This paper was presented at the *23rd Conference on Medical Image Understanding and Analysis* (*MIUA*) in 2019.
- "Automatically Localizing a Large Set of Spatially Correlated Key Points: A Case Study in Spine Imaging" [132] by Alexander Oliver Mader, Cristian Lorenz, Jens von Berg and Carsten Meyer. This paper was presented at the *22nd International Conference on Medical Image Computing and Computer Assisted Intervention* (*MICCAI*) in 2019.

1.4 OUTLINE

This thesis is comprised of four parts that are structured as follows.

The first part covers the problem definition and all related necessary information. In Chapter 2, an overview of the most relevant work related to this thesis w. r. t. object detection and localization is given. This briefly goes over very early parametric approaches to very recent model-free state-of-the-art approaches and how those compare to the work presented in this thesis. In Chapter 3, the necessary theoretical background on medical imaging and probabilistic graphical models is presented. Furthermore, a short explanation of decision tree (ensembles) and convolutional neural networks (CNNs) is given.

The second part introduces the method in great detail, as well as an alternative approach that is used for a thorough comparison later on. In Chapter 4, the general detection and localization approach based on a CRF is introduced, followed by the introduction of the used local appearance models (Chapter 5) and spatial models (Chapter 6). In Chapter 7, the proposed CRF optimization is introduced, which concludes our method. The last chapter of this part (Chapter 8) introduces the CNN-based method, which is later used for comparison.

The third part establishes the experimental setup that was used for the evaluation of the approach. In Chapter 9, the implementation of the method is described. This is followed by a listing of the six different datasets used for evaluation in Chapter 10 and a formal introduction of the evaluation metrics in Chapter 11.

The fourth part covers the detailed evaluation of the proposed method. In Chapter 12, we investigate in how far the proposed approach is applicable to different datasets and target key point configurations, while comparing the generated results to the results achieved by other state-of-the-art methods. In Chapter 13, we illustrate how the artificially reduced solution space—which is necessary for feasible CRF inference—might

reduce the localization performance and how our suggested "refine" label improves upon this. In Chapter 14, the proposed method is compared against the previously introduced CNN-based method while tackling the problem of localizing many key points in spine CT images in a reasonable amount of time. In Chapter 15, we investigate how the proposed method might influence other follow-up tasks—segmentation in this case—and how the previously introduced latent scaling variable compares to inherently scaling invariant ternary potentials. Finally, this part closes with our conclusions and suggestions for future work in Chapter 16.

2 RELATED WORK

In the following sections, various approaches tackling the problem of object localization are introduced and discussed. While focusing mainly on approaches being evaluated on medical datasets, noteworthy non-medical approaches are included as well, since a common pattern in medical imaging is that foundational ideas are pioneered on non-medical problems and transferred afterwards. Although the sections try to give a coarse classification of the approaches, it is not always possible to explicitly assign a method to one specific class as ideas are often also combined. Also note that here a broader and more relaxed definition of the object detection and localization task is applied, as often a more general method (e. g., bounding-box based detection which is often applied in computer vision) can be rephrased for a more specific task (here key point localization). Thus, such ideas are discussed as well.

While the first sections try to give a very much simplified outline of the development of object localization methods and the general trends, the latter sections focus on methods exploiting correlations between several key points. The last section compares the proposed method to the most aligned work and discusses the differences in detail.

2.1 PARAMETRIC AND TEMPLATED-BASED APPROACHES

An early notable approach for object localization is the so-called Hough transform [94, 61]. It allows for localizing analytic curves like lines or circles by constructing a dual parameter space from a binary edge feature image derived from a given digital image. Although limited to very few object types, this approach laid the foundation for an extension to arbitrary non-analytical shapes referred to as the generalized Hough transform (GHT) [8]. The GHT was well received in the scientific community and has seen various extensions and wide application in medical [214, 95, 115] as well as non-medical settings [14, 103].

Another template-based approach are correlation filters that are mostly used in tracking [22, 91] due to their high runtime performance. However, they have also been successfully applied in medical settings [179].

Also task-specific parametric approaches as in [126] have been proposed, although they are of less interest in this context due to their ad-hoc nature.

2.2 MACHINE-LEARNING-BASED APPROACHES

The further development of new object localization approaches was mainly driven by two major areas of research, the first being the development of compact representations of image characteristics in form of features. Ideally, these features are quick to compute, reduce the dimensionality and are robust against image transformations. This goes from simple intensity-based features (i. e., spectral information) over structural (e. g., edges [224] and corners [165]) and texture features (e. g., local binary patterns [146]) to invariant local patch descriptors (e. g., speeded up robust features [12]). For an exhaustive overview and comparison of different feature descriptors see [23]. The second area of research is the development of sophisticated machine learning methods in order to automatically train classifiers and regressors from data, where commonly the problem is cast to a regression or classification task and the data is given in form of feature descriptors. For example, support vector machines (SVMs) and decision trees—especially ensemble methods like random forests [25, 181, 145, 122], Hough forests [71, 123] and probabilistic boosting-trees [190, 36, 221]—have proven to be very successful in various domains including object localization in medical imaging [60, 6, 58, 77, 71, 51, 59, 78, 184, 148, 54].

Previous methods have also seen improvements using machine learning techniques. Consider for instance the template-based generalized Hough transform, which has been further extended using discriminative model combination and evaluated thoroughly in medical imaging [168, 167, 85, 86]. Furthermore, the simple object templates in form of shape edges have quickly been superseded by more versatile and discriminative means to capture the object structure. For instance, decision trees [147] and SVMs [217] have shown to improve the performance by assessing small local patches to capture the object structure and the relative correspondence.

Often, a global search for the target object is performed over the whole image domain. This however can result in an increased and unsatisfactory runtime, which is even more problematic in medical imaging given the (potentially temporal) volumetric high resolution data with eventually different aligned modalities (multi-channel). One way to counter increased runtime is to employ a coarse-to-fine search strategy [52, 167, 86, 223] by zooming into patches with different resolution. Another way is to use fast prior classifiers [126, 173, 181, 59, 97] to limit the actual search space to feasible regions. Yet another variant are specifically crafted features that allow to steer the search into a feasible subset as in marginal space learning [221, 102]. This is similar to approaches trying to regress displacements of feature vectors sparsely sampled from the image [37]. Note that the runtime is also heavily influenced by recent improvements in computing hardware, especially with the advent of

general-purpose computing on graphics processing units (GPGPU). Thus an evaluation of satisfactory runtime performance needs to happen per problem and needs to include the actual execution environment as well as a concrete method implementation.

2.3 NEURAL-NETWORK-BASED APPROACHES

The choice and development of features derived from the image data has a severe influence on the overall performance and hence is a crucial factor. However, with the recent development of artificial neural networks, especially convolutional neural networks (CNNs), and the increased availability and computing power of GPGPU, the problem has shifted. Successive layers in CNNs allow to automatically extract hierarchical feature representations, which can be used in later layers to, e. g., regress likely object positions. All model parameters—comprising the feature extraction and the regression part—can be learned in an end-to-end fashion from training data. Therefore, instead of manually crafting image features, the focus shifted towards exploring new CNN building blocks and their composition. In medical imaging, the focus is placed on estimating only few parameters aiming at high generalization in view of the mostly scarce amount of training data compared to, e. g., image classification tasks in general computer vision. This might be achieved by transfer learning strategies [180, 44] or by using models with fewer parameters.

The ideas vary vastly, from networks directly regressing coordinates [188, 183, 206, 114, 209] or displacements [216] combined with iterative refinement [119], over networks classifying voxel-wise direction quadrants [207] or region correspondence [97] to network combinations with prior feasibility classification [222]. Previous non-neural-network-based methods have also seen improvements by replacing older machine learning approaches with deep learning techniques as in deep marginal space learning [74] to replace the feature engineering or in deep reinforcement learning to realize the Q-learning in order to infer object detection actions [33, 99, 75, 76, 2, 197]. The end-to-end training philosophy has even been extended to discriminative correlation filters with connected feature extraction [179]. Also multi task learning has been successfully applied to object localization [112, 177, 208, 193, 119, 109], or similarly, as indirect by-product in a patch classification task [189]. However, the most dominant approach is the regression of pseudo probability maps over the image domain [150, 67, 112, 114, 149, 209, 20, 118, 175, 40, 151, 210, 223], given that convolutional neural networks are ideal for translation invariant reasoning in combination with readily available implementations for famous CNN architectures like the U-Net [164, 45, 98] or the V-Net [139]. The recent OBELISK-Net [90] tries to solve the problem of finding a reasonable receptive field

size [129]—which is dictated by the network architecture—using large sparse deformable convolutions while requiring fewer parameters.

2.4 INTEGRATING CONTEXT

Early on it has been evident that *context* plays a crucial role in robust object localization. This is even more important in settings with locally ambiguous objects, which is often the case in medical imaging. Consider for instance the left-right-symmetry of the human body or repetitive structures like the spine vertebrae or the finger joints. A reliable localization of such structures is often only possible by considering sufficient context, which is possibly constrained by a potential arbitrary, e. g., restricted, field of view.

Again, how context is considered and integrated varies in the different approaches. One approach is to ensure that derived features capture a certain spatial extent that is large enough to contain neighboring discriminating structures [221, 128, 108, 54]. Another approach is to learn the relative position of neighboring features [71, 123, 184, 149, 40]. The former one is analogous to ensuring that a CNN architecture has a sufficient effective receptive field [129, 90]. The notion of context in these approaches is implicit and embedded in the feature engineering or in the approach itself.

2.5 MULTIPLE OBJECTS

In case of *multi object* localization, the notion of context may also be viewed as dependencies between different objects and the correlation between those might be exploited. This is very reasonable, especially in the medical domain, given that objects are contained within the human body—a connected and mostly rigid structure with well-known kinematics—and that most tasks like measurements, registration, segmentation, etc. need predictions for multiple key points.

2.5.1 *Implicitly Exploiting Co-occurrences*

One general approach that acknowledges object dependencies is the auto-context model [191, 192, 161]. It is an iterative scheme in which the output of an object discriminator is fed as input into the next discriminator, in addition to the image. The key idea here is that successive discriminators can use the output of the previous discriminator as context cues and learn a proper selection. The manual feature engineering part (Haar features plus sparsely sampled classifier probabilities) and the discriminator have recently been replaced by a CNN [169] that allows to consider the full context, while not being trained end-to-end. However, the latter part has been illustrated in [162] by transforming

a regression-forest-based auto-context model [158] into a CNN with sparse convolutions, effectively allowing for end-to-end training and fine-tuning. This is similar to the two-step approach described in [195] using regression forests to first generate key point candidates, followed by an iterative coordinate descent step refining these positions using the candidates as well as geometric and appearance features.

The general auto-context idea has also been picked up in the convolutional pose machine (CPM) architecture [203, 20], in which dense convolutions are used with intermediate supervision after each stage (iteration) to form a cumulative loss objective for end-to-end training. Whether the network actually learns a constellation in successive stages is not mandated, unlike in networks with a separation between local appearance model and constellation model, where the latter has only access to the local appearance predictions (heatmaps) and not the image itself, like in [154, 150, 151].

A different approach is followed in [43, 175, 193, 42] by using a conditional generative adversarial network where the discriminator is used to tell correct from incorrect constellations apart. Due to the adversarial training of the generator, the constellation prior is implicitly exploited.

Recently, object detection reinforcement learning has been adapted to detect multiple objects by using multiple agents with implicit interactions [197]. During training, the agents share their knowledge (in form of shared network weights) to obtain a collective gain. Note that the notion of context here is different to most other approaches, since it includes the image information but also the collective reasoning w. r. t. the agents. Another approach is followed in [99], where tree-structured reinforcement learning is used to train an agent to sequentially detect multiple objects by considering the search path.

A less often applied approach is multi-atlas registration as in [66] for key point labeling, as it is computationally more involved and targeted towards rigid structures.

2.5.2 *Explicit Dependency Modelling*

Another very large branch of methods tries to explicitly model the dependency between objects. As mentioned earlier, this is a reasonable idea in medical imaging, where most tasks require multiple well-defined localization predictions within a mostly rigid body.

One well known approach pioneered in the field of object shape modelling are the active shape models (ASM) by Cootes et al. [48, 92, 49, 50]. In ASM, an object's shape is represented by a set of well-defined landmark positions. Using principal component analysis (PCA), a statistical shape model referred to as point distribution model (PDM) can

be derived from a set of such annotated training images, allowing to generate object shapes found in the training set distribution. In combination with point-specific image appearance models, strong edges in this case (cf. active contour models like snakes [100]), an iterative process is used to transform the shape parameters of the PDM to match a given image best until no significant changes occur anymore. The rather limited edge features have been overcome by the successive method called active appearance models (AAM) [46, 47, 50]. The target object is projected into a shape-normalized frame in order to model the grey-level appearance in a normalized image region w. r. t. the shape model. Note, this might be used to perform multi-object localization as well just by treating the object as a super-object, which contains the actual objects of interest whose positions are used for shape modeling. See also constrained local models [123, 122] as a class of methods localizing key points constrained by a statistical shape model. There is a zoo of other deformable-template-based approaches, but a key insight from these contributions is that shape modelling—i. e., encoding an objects position in relation to some higher level context—helps the automatic interpretation even in noisy or cluttered images or where objects may be occluded. See for example [123, 209, 119, 210] for object localization approaches exploiting a shape prior.

Another well known approach are deformable part models such as pictorial structures. Introduced 50 years ago by Fischler et al. [69], the idea has been picked up by Felzenszwalb et al. [65, 64] and enhanced by part-specific discriminative models connected by "springs". Note that the different parts are treated as latent and automatically learned. For a supervised application see the mixture of parts extensions by Yang et al. [211] to articulated human pose detection. These ideas are similar to the constellation model idea by Weber et al. [202] where clustering (unsupervised) is used to determine relevant parts, which is based on the supervised and principled approach by Burl et al. [31] using probabilistic shape models in combination with the responses of the individual part detectors. Again, these methods illustrate the benefit of understanding an object as a combination of interacting parts, while improving upon holistic template-based approaches such as [119, 210].

Another branch of methods [155, 17, 60, 172, 7, 16, 58, 77, 9, 59, 78, 41, 37] uses probabilistic graphical models (PGMs) as spatial regularization, commonly seen in form of a CRF [113]. In medical imaging, such models can be used to solve the confusion between locally ambiguous target objects like the spine vertebrae [172, 58, 77] or the finger joints [60]. The rich modelling capabilities in combination with a principled theoretical foundation render PGMs ideal to model part-based detection problems [200]. For instance, the previously discussed pictorial structures can be naturally modeled as a part-based Markov

network. In [65], a tree-like Markov network was used to model spring-like priors through binary potentials in combination with unary data likelihood potentials.

The main problems in graph-based part dependency modelling are the definition of a reasonable graph topology in combination with suitable knowledge sources represented by potential functions. Often, the topology is chosen manually derived from the underlying anatomical structure, e. g., a chain topology for the spine vertebrae [78] or a tree structure for human joint modelling [41]. This is often done to allow for linear inference time, but might potentially miss important interactions. In contrast, in [16] a fully connected graph, i. e., the opposite extreme, is used. It might include redundant information while additionally increasing the inference time due to the highly cyclic graph topology. In [60, 58], Delaunay triangulation is used to connect locally neighboring key points to build a sparse connectivity model. This, however, is an uninformed heuristic and creates a locally cyclic topology. In [59], a heuristic based on the differential entropy of the distribution of relative key point distances is used to derive the topology by looking for pairs with low entropy and thus little variance in their relative positions. However, this may not be optimal if other features than the relative distance are used to characterize the key point pairs. This problem also applies to the statistical measures evaluated in [163]. Another class of structure learning algorithms utilizes a set of conditional independence tests (also known as constraint-based) as in [27] to find a structure consistent with the outcome of the individual independence tests. Although the assumption of the correctness of each test has been relaxed in [171], this search is still considered $\mathcal{NP}$-hard and is generally used in knowledge discovery rather than prediction tasks [106].

The second problem of defining suitable knowledge sources is split up into defining part-specific (unary) potential functions and constellation-evaluating (non-unary) potential functions. Commonly, machine learning approaches like random forests [59, 78] or neural networks [41, 9] are used to build feature detectors to derive unary potential functions from describing the likelihood of a key point being located at a specific position. However, ad-hoc methods utilizing dedicated feature detectors are used as well [16]. Higher order potential functions are then in turn used for spatial regularization. Often, binary (pairwise) potential functions are employed, computing simple statistics on spatial properties like distances [16, 59, 78, 4, 89], angles [16], epipolar constraints [16], spatial constraints [198], etc. However, also image features in form of gradient vector fields [60], edge separation [4], surface normals [4], image patches [16, 41], etc. are used. Admittedly, these image features are used less often due to the increased computational complexity. As most binary potentials are often not both rotation and scaling invariant, if even just one of both, ternary potentials can be

formulated to elegantly pose those properties while at the same time increasing the computational complexity. For example, in [198] ternary potential functions construct an internal coordinate system to be rotation and scaling invariant. Alternatively, latent variables can be used to model global variation, consider for example the pose in [157]. Overall, the definition and usage of potential functions is mostly heuristically motivated.

Object dependencies have also been modeled explicitly in neural-network-based approaches. For example, the multi-architecture deep image-to-image network by Yang et al. [209] uses a recurrent neural network (RNN) in form of a long short-term memory (LSTM) network as post-processing step to enforce the chain structure of the spine vertebrae. Message passing (or belief propagation) inference from the field of PGMs also found its way [220, 140, 219] into neural networks, allowing for approximate inference in arbitrary part connections, unlike the LSTM chain structure. A different approach is followed in [206], where a dependency matrix is used to encode dependencies between regressed positions to derive the final position from.

2.6 DISCUSSION

The amount of methods using artificial neural networks is steadily increasing, which can generally be attributed to their superior performance in various tasks in combination with readily available open source software in this area of research. This is also the case in general object localization, where dedicated network architectures with modules to exploit the spatial correlation between landmarks seem to outperform all previous methods [203, 209, 20, 151]. However, the excellent performance of neural networks is often paid for with an intransparent prediction process. Although there are flows towards explainable deep learning models [3, 170], this is still a niche in applied deep learning and in current state-of-the-art approaches.

It is however, especially in the medical domain, important to know how a method arrived at a conclusion and what the limits of that method are [143]. For instance, it is unclear what kind of dependencies are learned in deep-learning-based approaches like [203, 151, 197], especially since it is often not possible to train a whole image (imagine a large high resolution CT volume with many target key points) at once requiring patching strategies that potentially miss important long range dependencies. In contrast, using probabilistic graphical models (PGMs) allows for modelling object dependencies in a flexible and theoretical sound way while allowing to incorporate all kinds of knowledge sources like shape models and neural networks. As we have seen earlier, it is possible to reformulate different approaches as PGMs, demonstrating their flexibility and general applicability.

With respect to object localization however, only few methods utilizing a PGM—notably the method by Donner et al. [59]—provide a general and easy to transfer setup achieving state-of-the-art performance in medical imaging. The two key aspects of defining the graph topology and selecting suitable potential functions are often solved heuristically, making it hard to transfer the approach to new datasets. In contrast, the approach proposed in this thesis allows to easily estimate the important graph parameters from data using a criterion related to the detection and localization task. The key is to define a *pool* of candidate potential functions and to *weight* the used potential functions in order to remove unnecessary zero-weighted ones. Starting from a fully connected and fully loaded graph, most relevant potential functions are automatically selected while concurrently optimizing the topology and thus the runtime. In addition to optimizing the topology, the potential fitness is captured in the weight as well, similarly to [218] but for all potentials.

The detection problem is solved in a principled way by introducing a special "missing" label as done in [16], but instead of heuristically estimating the corresponding potential energies they are learned jointly with the potential weights. Although only spatial non-unary terms are being used, any other potential function as discussed earlier (e. g., image data depending ones) might be used as well and the approach automatically selects the suitable ones. To do so, an energy-based max-margin formulation [117] is optimized using stochastic gradient descent. This allows to ignore the computational involved partition function as in [16], where maximum likelihood estimation (MLE) is used to optimize the parameters and additional true configurations are introduced to relax the MLE criterion. This however becomes intractable quickly and requires special weighting to not stress the influence of outliers. In contrast to [107], we do not resort to approximate inference approaches, although that might be possible in case the computational complexity is becoming infeasible. In [9], the potential functions are fine-tuned jointly w. r. t. the graph topology, while here they are optimized independently. However, the learning framework can be easily extended to jointly optimize the potential functions as well, given that they are differentiable w. r. t. their parameters, which is not the case for all types of potential functions.

3 THEORETICAL BACKGROUND

In order to properly understand the proposed method and the successive evaluation, the necessary theoretical background on medical imaging, decision trees, artificial neural networks and probabilistic graphical models is presented in the following sections. A reader familiar with the concepts may safely skip these sections, which are independent of each other and may be consumed selectively.

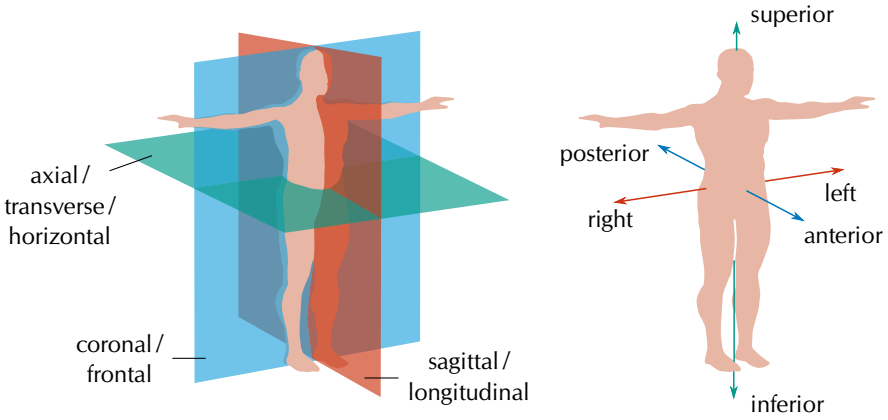
3.1 MEDICAL IMAGING

Medical image processing is different to the analysis of natural images, since the object of interest being imaged is often found inside the human body. Firstly, this requires different image acquisition techniques that are able to "view" into the human body. Secondly, in order to properly locate the human body (or a part of it) inside such an acquired image, special terms to describe the location and orientation are used that form a coordinate system w. r. t. the imaged anatomy.

3.1.1 *Anatomical Location*

In the human anatomy, a set of well-defined terms—mostly in Latin and Greek—is used to precisely communicate a location w. r. t. the human body. The most important terms for this work are depicted in Fig. 3.1. This includes illustrations of the three human body planes coronal, sagittal and transverse (Fig. 3.1a) and of the directions posterior–anterior, left–right and inferior–superior (Fig. 3.1b). A combination of the directional terms is also used to describe an axis, e. g., the superoinferior axis going from the head towards the feet of a human. Often, the planes are also used to describe an axis, which corresponds to the normal of the plane. For example, the sagittal axis corresponds to the left-right axis. For a thorough introduction to the human anatomy and standardized terms describing the anatomical location see [187].

In contrast to conventional imaging, most medical acquisition techniques provide a correspondence between image coordinates and physical coordinates. Thus, it is often possible to establish a physical world coordinate system w. r. t. the acquisition device in which the acquired image might be placed. This is encoded in image meta data describing the origin of the image (in the world coordinate system), the physical spacing between pixel / voxel centers and the directions of the image axes. In addition to this world coordinate system, it is often useful to



(a) Planes (b) Directions / axes

Figure 3.1: Illustration of common medical terms to describe an anatomical location in terms of (a) planes and (b) directions. Note that combinations of directions are used to describe an axis, e. g., the posteroanterior axis is going from the back of a human to its front. Based on [141].

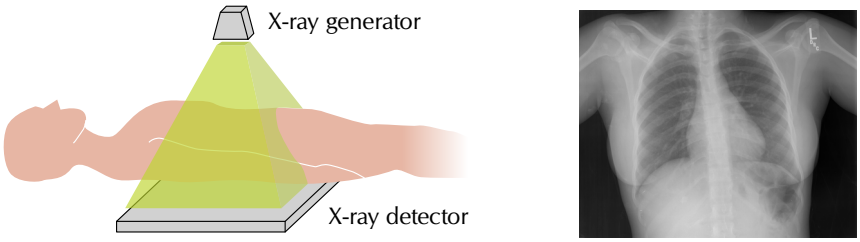
describe the location of the human—or a part of it—inside that acquired image and establish correspondences between the human axes and the image axes. This is encoded (if known) in either the image meta data or as a convention of the used image format.

3.1.2 *Image Acquisition*

The imaging modality and thus the image acquisition technique dictates image parameters as its dimensionality, spatial resolution and value interpretation. In the following, the peculiarities of the image modalities used in this work are briefly described. The interested reader is referred to [201] or [19] for a more thorough overview.

Projectional Radiography (X-ray)

The conventional form of radiography uses X-radiation to create an attenuation image of the imaged anatomy, simply referred to as X-ray. The imaged anatomy is placed between an X-ray generator and a corresponding detector, which is either a photographic screen (analog) or nowadays a flat-panel detector (digital). The generator produces an ionizing X-radiation that is directed towards the detector. On its way, the X-ray passes through the imaged anatomy and is attenuated. The attenuation is more efficient in dense matter (e. g., bone or metal implants) and less so in soft tissue. Thus, the contrast in the resulting digitized 2D single-channel image arises from the differential attenuation. See Fig. 3.2 for a schematic illustration in combination with a potentially resulting example chest radiograph.



(a) Schematic structure (b) Chest radiograph

Figure 3.2: (a) Simplified schematic illustration of a projectional X-ray radiography setup and a potentially resulting (b) chest radiograph. Note the darker areas with less attenuation in the lung field due to the air and the stronger contrast of the rib cage surrounding it. Based on [201, Fig. 1.1].

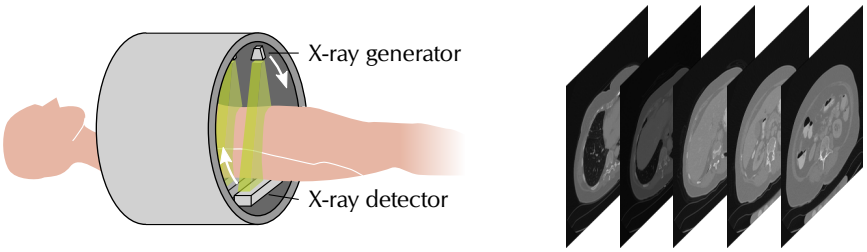
The resulting image values are not normalized and depend on the used dosage and the matter traversed. Furthermore, there is no direct correspondence between the image size and the physical extent, as the object is enlarged due to the distance between the object and the detector. Thus, the resolution of a pixel (px) is not known and has to be estimated by, e. g., an expert or structural comparisons.

Despite the astonishing possibility to view into the human body, care has to be taken as this method uses ionizing radiation, which may damage tissue cells. Thus, there is a yearly dosage limit a patient may be subjected to. For simplicity, we refer to this imaging technique simply as X-ray.

Computed Tomography (CT)

The same principle of an X-ray source generating a beam traversing through the imaged anatomy is also used in computed tomography (CT). In order to generate a cross-sectional image (slice), multiple projectional line radiographs at different angles around the imaged anatomy are generated. Then, the inverse Radon transform can be used to compute the 2D tomographic image from the set of line radiographs collected over different angles, each one corresponding to a set of line integrals represented by the total attenuation of an X-ray beam traversing the object in a straight line at that angle; this process is referred to as "reconstruction". The 2D representation of the line radiographs is called a sinogram. By generating multiple slices and stacking them, a single-channel 3D image of the imaged anatomy is created. See Fig. 3.3 for a schematic illustration in combination with potentially resulting CT slices. Note that we refer to X-ray CT when talking about CT, but want to mention that there exist other types of CT such as positron emission tomography (PET), which are not used in this work though.

In contrast to projectional radiography, the physical location of the scanned anatomy w. r. t. the scanner is known, thus a world coordinate



(a) Schematic structure (b) Torso CT slices

Figure 3.3: (a) Simplified schematic illustration of a computed tomography (CT) scanner and potentially resulting (b) axial torso CT slices, which are stacked to form a 3D image of the torso. The illustration in (a) is based on [201, Fig. 1.2].

system can be constructed. Commonly, the in-plane (slice) resolution is higher than the out-of-plane (slice distance) resolution, resulting in anisotropic (non-cubic) voxels (vx). This can be countered by thinner slices or modern scanners with isotropic acquisition.

Additionally, the value range is normed using the Hounsfield scale. The attenuation coefficient μ of a voxel is linearly transformed into Hounsfield units (HU) by fixing water and air at 0 HU and −1000 HU, respectively:

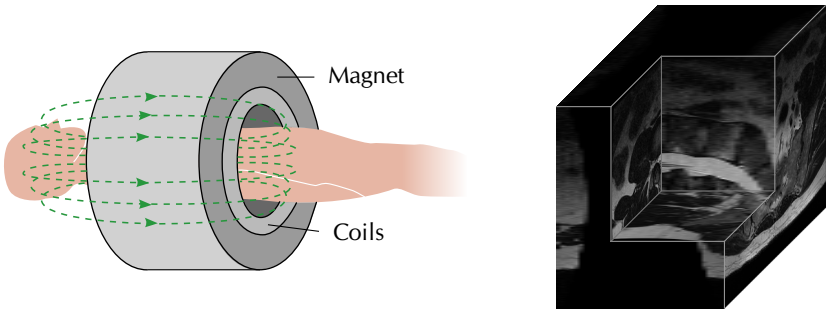
$$hu(\mu) = \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}} \cdot 1000\,\text{HU}\,. \tag{3.1}$$

Although the range does not impose an upper limit, a value range from −1024 HU to 3071 HU encodable in 12 bit is often used in practice. Due to the Hounsfield scale, the range of image values corresponding to a given tissue type is generally comparable across different scanners and dosages.

Similar to projectional radiography, the usage of ionizing radiation poses a health risk as it may damage the tissue. This is in CT even more true, as multiple projectional radiographs are generated increasing the subjected dosage.

Magnetic Resonance Imaging (MRI)

A less harmful acquisition technique is magnetic resonance imaging (MRI), which provides better soft tissue contrast while needing more time to acquire an image. It is based on the fact that hydrogen nuclei aligned to a magnetic field emit a radio frequency signal when they are forced out of equilibrium and go back to the equilibrium state. To do so, the patient is placed in a large magnetic field, causing the hydrogen nuclei to align with that field. Next, a radio frequency pulse is applied, causing the hydrogen nuclei to tilt away from the magnetic field. Once the radio frequency pulse is removed, the nuclei realign



(a) Schematic structure (b) Lower torso MRI volume

Figure 3.4: (a) Very simplified schematic illustration of a magnetic resonance imaging (MRI) scanner and a potentially resulting (b) lower torso MRI volume, which has been artificially incised for better illustration. Note the highly anistropic (non-cubic) voxels with a resolution of $0.39 \times 0.39 \times 4.5\,\mathrm{mm}^3/\mathrm{vx}$, which can be seen in the difference of sharpness of the different image planes.

themselves with the magnetic field to reach equilibrium. During that process, referred to as relaxation, the nuclei lose energy in form of a radio frequency signal that is picked up by a corresponding coil and measured over time.

A gradient magnetic field is added causing a linear axial frequency change that allows to "select" an axial slice for reconstruction. Within each axial slice, the spatial reconstruction is achieved using frequency and phase encoding using repeated excitations and relaxation measurements. Thus, a world-coordinate system w. r. t. the scanner can be established. Note that the in-plane resolution is often higher than the out-of-plane resolution and that the spatial resolution is in general worse than in CT. In Fig. 3.4, a very simplified schematic illustration of an MRI scanner in combination with a potentially resulting MRI scan is shown.

The resulting image values are not normalized and depend on a large number of factors. Important ones are the used scanner and the imaging protocol, which corresponds to different physical properties being evaluated. The choice of protocol mostly depends on the medical application and the anatomy being measured, i. e., different properties to highlight. See Fig. 3.5 for an illustration of different MRI modalities in terms of the resulting image, which corresponds to a multi-channel 3D image in this case. This increases the problem of comparability between scans from different scanners and different sites.

In contrast to CT, this is a much safer imaging modality for the patient as it does not use ionizing radiation.

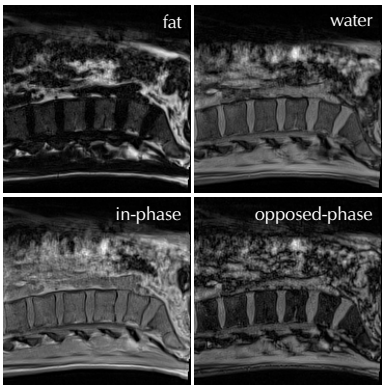


Figure 3.5: Sagittal slices of four MRI modalities acquired from the same anatomy using an 1.5 T Siemens scanner and the Dixon protocol [57].

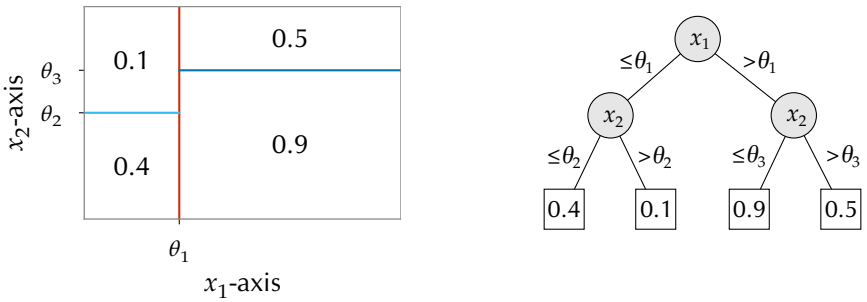
However, the time to acquire an MRI scan takes much longer and not every patient is able to rest for such prolonged time in an uncomfortable wedged position with a lot of loud mechanical noise.

This concludes the introduction to medical image acquisition (needed for this thesis) and we continue with the different machine learning concepts used in this thesis.

3.2 DECISION TREES

A simple method of describing an iterative decision process based on a multidimensional input space are so-called *decision trees*. Often, a convenient representation in form of binary trees is used where their traversal corresponds to the iterative decision process: At each non-leaf node of the tree, a simple model performs a binary decision based on the value of (often just) one input axis and thus dictates the branch—left or right—to be followed. This process is repeated until a leaf node is reached, which provides the final outcome. Depending on the type of leaf node, decision trees can be used to model classification tasks as well as regression tasks. For example in case of the former, one common mode of operation is to compute the probability of an input vector belonging to some class. While this is a common setup found in practice, note that non-binary trees (breadth instead of depth) as well as decisions based on more than one axis (oblique trees) are used as well.

This decision process corresponds to a partitioning of the input space into cuboid regions. Commonly, the edges of the region are aligned with the axes and correspond to a decision based on the respective value. An example of such a partitioning including the decision tree representation is illustrated in Fig. 3.6 for a two-dimensional real-valued input space. [21, pp. 663–666]



(a) Input space partitioning (b) Tree representation

Figure 3.6: Example of a two-dimensional input space that (a) has been partitioned into four regions in combination with (b) the corresponding binary decision tree representation. Based on [21, Fig. 14.5 and 14.6].

3.2.1 *Construction*

The construction of such decision trees is an example of supervised machine learning, i. e., the goal is to construct a model from training samples where each sample dictates the desired output for a given input vector. However, constructing a globally optimal tree is known to be an $\mathcal{NP}$-complete problem, which is why practical algorithms are based on heuristics. One early framework are the classification and regression trees—also known as CART—by Breiman et al. [26]. However, there are also other variants such as ID3 by Quinlan [159], or its successor C4.5 [160]. Here, we focus on the first.

Generally, a greedy algorithm is used to perform locally optimal decisions in order to circumvent the combinatorial complexity and to recursively grow the tree starting from a root node (which provides the whole input space). At each step, a joint exhaustive optimization is performed to decide which input variable to use and which threshold to pick using a specific splitting criterion. In contrast to the optimization over the whole tree, this local optimization can be done efficiently. The process is continued at all nodes until some stopping criterion is fulfilled.

Depending on the type of tree—classification or regression—different types of splitting criterion as well as stopping criterion have to be used. In case of regression, a commonly used splitting criterion is *variance reduction*

$$I_V = V(\mathcal{S}) - \left[V(\mathcal{S}_{\leq}) + V(\mathcal{S}_{>})\right], \tag{3.2}$$

which computes the variance

$$V(\mathcal{S}) = \frac{1}{|\mathcal{S}|^2} \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} (y_i - y_j)^2 \tag{3.3}$$

of the target value y for the samples before the split $\mathcal{S}$ and after the split for both branches $\mathcal{S}_{\leq}$ and $\mathcal{S}_{>}$. In case of classification, Gini impurity or information gain are commonly used measures.

Using these measures, a tree is generally grown until a threshold is met, a certain amount of minimal samples is reached or a certain tree depth is reached. While the latter two already help to increase the generalization, the tree is pruned after it has been fully grown by removing leaf nodes in order to reduce the model complexity in favor of better generalization, accepting a potentially higher variance.

A major strength of decision trees is their speed in terms of training as well as test time in combination with the simplicity of the model. While the white-box nature allows to understand the reasoning of a tree (in contrast to a black-box method), careful measures have to be taken

to not overfit the tree onto the training data. For more information on how to create a decision tree from training data see the aforementioned literature.

3.2.2 *Ensembles*

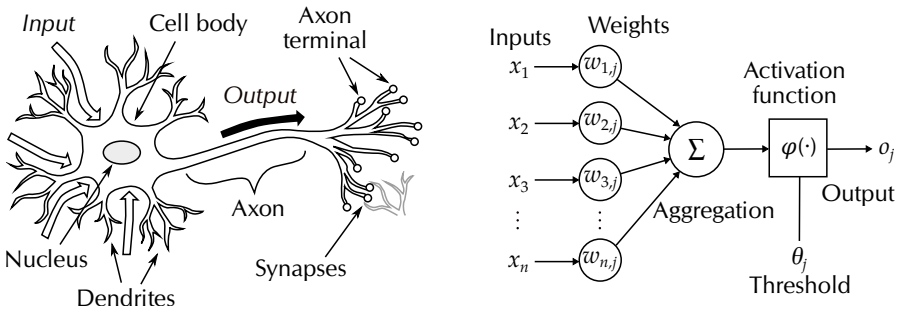
While there are different means to improve the performance of a decision tree, for instance the usage of non-axis-aligned splits resulting in oblique trees, a more common approach is to form an *ensemble* of multiple trees. The main idea is that the deficits of a member of the ensemble are cancelled out by the other members, given that they learned different characteristics.

A popular meta-concept is called *boosting*, which iteratively creates weak learners—e. g., small decision trees—where the successive learner focuses on the errors of the previous one and all form a weighted ensemble. A well-known example is AdaBoost by Freund and Schapire [70].

Another concept is called *bagging* (bootstrap aggregating) introduced by Breiman [24], which creates multiple bootstrapped training sets by resampling the training data with replacement and learning a predictor for each one. The final result is the combination of all predictors, which forms an average in case of regression. A special case of the latter concept in the context of decision trees are random forests (RFs) [25]. In addition to performing the training set resampling, the node split is performed on a *random subset* of the available features in order to further de-correlate the trees. Generally, the use of randomness as means to form decision tree ensembles has shown to achieve improved results [73].

3.3 ARTIFICIAL NEURAL NETWORKS

A well-known machine learning approach that got much attention over the recent ten years (although it existed much longer) are the so-called artificial neural networks (ANNs). They are vaguely inspired by neural networks within the human brain, where interconnected *neurons* form a neural network. See Fig. 3.7 for a simplified illustration of a biological neuron as well as a derived artificial neuron. Note that both have various inputs (connected to the outputs of other neurons) and produce an output themselves. Thus, the mathematical formulation of the depicted artificial neuron $o_j = \varphi(\sum_{i=1}^{n} x_i w_{i,j} + \theta_j)$ is a weighted summation of n inputs, which is gated by an activation function like the step or sigmoid function. In this setting, the bias θ_j represents a threshold, which mimics the electrical potential in a biological neuron that has to be reached (aggregation) in order to generate a pulse along the axon. [84]



(a) Biological neuron (b) Artificial neuron

Figure 3.7: (a) Simplified illustration of a biological neuron w. r. t. the aggregation of input signals (from other neurons) to form an output signal (to other neurons), based on [18, Fig. 4.19]. (b) Illustration of a common artificial neuron, which is a weighted aggregation of inputs gated by an activation function, based on [30].

3.3.1 *Types of Networks*

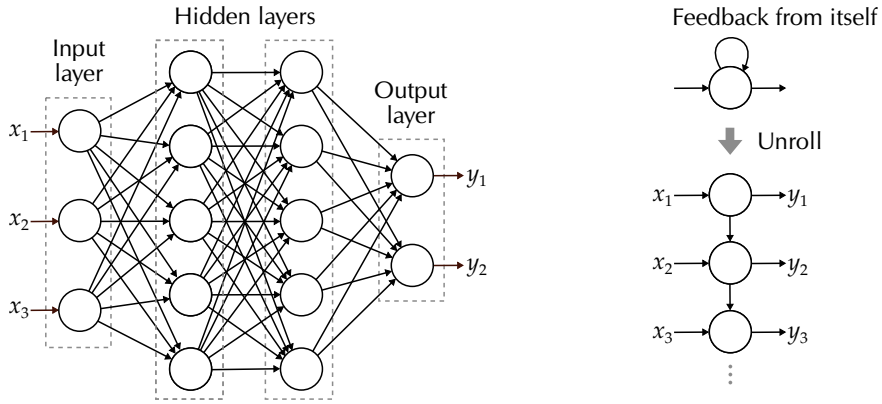
An artificial neuron just by itself is not sophisticated enough to model any interesting real-world problem (it is just a linear discriminator, also known as *perceptron*), but the composition of such neurons in networks allows to model complex tasks. Generally, ANNs are classified as either *feed-forward* neural networks or *recurrent* neural networks (RNNs). [79]

In the former, information flows from the input through some neurons to the output without any feedback loops. Commonly, these neurons are organized in *layers*: one input, one output and multiple hidden layers. Hence, such a setup is also known as the *multilayer perceptron*, the goal of which is to approximate some function f^* by mapping an input vector $\boldsymbol{x}$ to the output domain $\boldsymbol{y} = f(\boldsymbol{x}; \boldsymbol{w})$ w. r. t. the set of tunable weights $\boldsymbol{w}$. Depending on the type of the output layer, classification as well as regression tasks can be modeled. [79]

In RNNs, the neurons additionally incorporate feedback from themselves, which allows for *dynamic* behavior. This is often used for modeling (temporal) sequences, as is done, e. g., in the well-known long short-term memory (LSTM) architecture. Both network types are illustrated in Fig. 3.8. In the following, we focus on feed-forward neural networks, because they specialize in the processing of grid data such as images. [79]

3.3.2 *Parameter Estimation*

Assuming a given network architecture, a central question is how to estimate the missing parameters (e. g., weights $\boldsymbol{w}$) in order to approximate the function f^* best. In a supervised setting, it is common to



(a) Feed-forward multilayer perceptron (b) Recurrent neural network

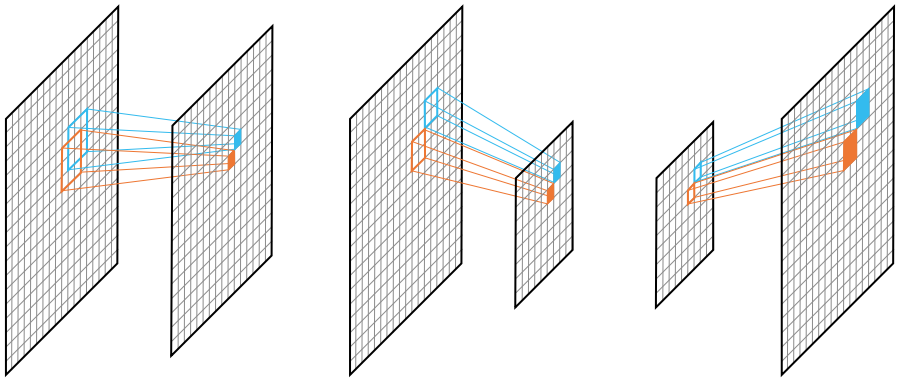
Figure 3.8: Exemplary Illustration of a (a) feed-foward neural network in terms of a multilayer perceptron and a (b) recurrent neural network that incorporates feedback from itself.

define a *loss function* (also known as cost function) that quantifies the discrepancy between the network's output $\boldsymbol{y}$ and the respective true output $\hat{\boldsymbol{y}}$ for a given input $\boldsymbol{x}$. The parameter estimation problem can then be formulated as minimization of the loss function, which is a (continuous) surrogate for an error function which may be discrete in case of a classification task. Since the loss function is differentiable with respect to the parameters, optimization techniques such as *gradient descent* can be applied to minimize the loss function, which then yields an estimate for the parameters. Note that gradient descent only finds the global minimum in case of a convex loss function, which is often not the case and hence the found minima are local ones. As an example, in regression a commonly used loss function is the sum of squared errors (SSE) loss $l = \sum_i (y_i - \hat{y}_i)^2$.

In order to compute the gradients necessary for the optimization efficiently, the *backpropagation* algorithm has been proposed by Rumelhart et al. [166]. While it is straightforward to compute an analytical expression for the gradient, the computation of it often expensive, especially in deep networks. The backpropagation algorithm uses the chain rule to compute the gradient of the loss function w. r. t. each weight. After a forward pass through the network to obtain the network error and thus the loss value, the gradients are iteratively computed going backwards through the network—layer by layer—in order to avoid redundant computations. This is done iteratively, updating the parameters in each iteration using the gradients computed and averaged for a set of training samples. The set of training samples used in each iteration is referred to as *mini-batch*. In case the mini-batch contains only 1 training sample, this corresponds to online learning, while using all training samples corresponds to batch learning. However, it has been observed that the usage of a mini-batch size that lies in between those two extremes is ben-

eficial in terms of time to convergence and the resulting generalization capability. Note that the backpropagation algorithm is a special case of *reverse automatic differentiation*, which generally allows to efficiently and accurately generate numerical derivative evaluations by accumulating values during code execution at machine precision [13]. In combination with general-purpose computing on graphics processing units (GPGPU) and the ever increasing performance of graphics processing units (GPUs), this allowed to train larger and deeper networks on very large datasets efficiently.

3.3.3 *Convolutional Neural Networks*



(a) Convolution (b) Pooling (c) Un-pooling

Figure 3.9: Illustration of three major receptive-field-adjusting operations used in encoder-decoder architectures in terms of modifying a single feature map (e. g., a grayscale input image): (a) A 3×3 convolution without padding, hence the slightly reduced output size. (b) A 2×2 pooling operation with a stride (step size) of 2×2, note the highly reduced output size. (c) A 2×2 un-pooling operation to upsample back to the original size, i. e., the input size in (b).

A special kind of feed-forward neural network is a convolutional neural network (CNN), which represents a dominant architecture type in modern computer vision. In contrast to *fully connected* layers, the layers of CNNs make use of *parameter sharing* in combination with a *local connectivity pattern* based on the grid. This is equivalent to convolving, more precisely cross-correlating, the output of the previous layer—generally referred to as feature map—with a fixed-size, small kernel, which is equivariant to translation in contrast to the fully connected layers and greatly reduces the amount of parameters when the kernels are small. The convolution is often followed by a non-linear activation function like the rectified linear unit (ReLU) $\varphi(x) = \max(x, 0)$. Generally, it has been observed that using more layers with small kernels performs better than fewer layers with larger kernels, hence the aim for deeper networks [116]. The intuition behind this is the hier-

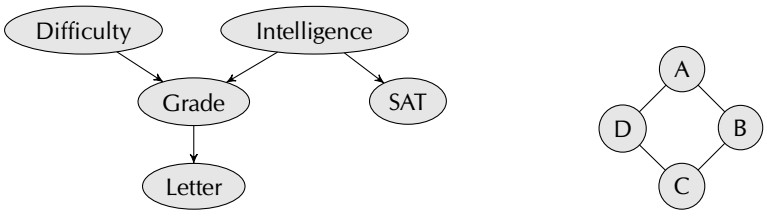
archical modeling of features with increasing complexity, e. g., going from simple edge filters to complex object representations.

The *receptive field size* is a driving factor when designing such a CNN architecture. It specifies how much of the input image has been seen by an output neuron. Note that the theoretical receptive field size might be larger than the effective one [129]. Depending on the task, this size might need to be different and potentially as large as the input image. In order to quickly increase the receptive field size, *pooling operations* are often used, which change the feature map resolution by "downsampling" and thus greatly increase the receptive field size of successive convolutional layers. In order to undo this downsampling, operations such as a *transposed convolution* can be used. Encoder-decoder architectures [212] make use of these operations by first creating an abstract downsampled latent representation before upsampling back to the original resolution. Some common operations are depicted in Fig. 3.9 (see later Fig. 5.3 for a full encoder-decoder architecture). For a detailed arithmetic explanation of common layer types in CNNs see, e. g., [62].

3.4 PROBABILISTIC GRAPHICAL MODELS

A probabilistic graphical model (PGM) is a probabilistic model of a complex system in which a graph-based representation is used to compactly encode a complex distribution over a high-dimensional space; also referred to as *graphical model*. The nodes in such a graph correspond to random variables while edges between them correspond to probabilistic interactions between them. The conditional dependence structure is directly expressed in the graph. [106]

3.4.1 *Types of Graphical Models*



(a) Bayesian network (b) Markov network

Figure 3.10: Illustration of the graph-based representation of the two types of graphical models (a) Bayesian networks and (b) Markov networks using two example graphs. The example in (a) is based on [106, Fig. 3.3].

There are two common types of graphical models that allow to encode different kinds of conditional independencies and thus different families of distributions. The first family are *Bayesian networks* (also referred to as

belief networks), which are often used for causal reasoning and hence represented by directed graphs. Thus, they are also known as directed graphical models. The second family are *Markov networks* which are often referred to as Markov random fields (MRFs) and represented by undirected graphs, hence they are also known as undirected graphical models. See Fig. 3.10 for graphical representation of both types.

A major strength of graphical models is the perspective of compactly representing a potentially high-dimensional distribution using *factors*. Instead of encoding the probabilities of all possible value assignments over all random variables, based on (conditional) independence relations the distribution is split-up into factors each covering a much smaller scope and thus smaller space of probabilities. The overall distribution is then given by the product of factors. Note that the joint distribution satisfies the conditional independence properties encoded in the graph representation.

In the following, we focus on *discrete* Markov networks and refer to the probability mass function as probability distribution.

3.4.2 *Markov Networks*

A Markov network is represented by an undirected graph denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of the set of nodes $\mathcal{V} = [1\,..\,N]$ and the set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Each node $i \in \mathcal{V}$ is associated with a random variable denoted by X_i, with its value x_i taken from the discrete *label* space $\mathcal{X}_i$ (i. e., $x_i \in \mathcal{X}_i$). Let $\boldsymbol{X} = \{X_i\}_{i \in \mathcal{V}}$ denote the indexed set of joint random variables and $\boldsymbol{x} = \{x_i\}_{i \in \mathcal{V}}$ the indexed joint values taken from the *state* space $\mathcal{X} = \mathcal{X}_1 \times ... \times \mathcal{X}_N$ given by the Cartesian product of all label spaces. The direct neighbors of the i-th random variable are given by the direct neighborhood $\mathcal{N}_i = \{\, j : (i,j) \in \mathcal{E}, j \neq i \,\}$ also referred to as *Markov blanket*. Note that the Markov blanket in Bayesian networks also includes indirect neighbors.

Independence Assumptions

For the random variables $\boldsymbol{X}$ to form a Markov network with respect to $\mathcal{G}$, three sets of independence assumptions must hold. The notation $\langle A \perp\!\!\!\perp B \mid C \rangle$ is used to state conditional independence between A and B given C.

PAIRWISE MARKOV PROPERTY Two non-neighboring random variables are conditionally independent given all other variables:

$$\{\, \langle X_i \perp\!\!\!\perp X_j \mid \boldsymbol{X}_{\mathcal{V} \setminus \{i,j\}} \rangle : i \in \mathcal{V}, j \in \mathcal{V}, j \notin \mathcal{N}_i \,\}. \tag{3.4}$$

LOCAL MARKOV PROPERTY A random variable is conditionally independent of all other variables given its neighborhood:

$$\{ \langle X_i \perp\!\!\!\perp \boldsymbol{X}_{\mathcal{V} \setminus \mathcal{N}_i \setminus \{i\}} \mid \boldsymbol{X}_{\mathcal{N}_i} \rangle : i \in \mathcal{V} \}. \tag{3.5}$$

These independence assumptions can be viewed as a reachability problem in the graph $\mathcal{G}$. The independence assumption $\langle A \perp\!\!\!\perp B \mid C \rangle$ states that there is no path from A to B that does not contain C. Thus, it is said that C *separates* A and B indicated by $\operatorname{sep}_{\mathcal{G}}(A, B \mid C)$. This notion of separation can be used to state the last property:

GLOBAL MARKOV PROPERTY A set of random variables is conditionally independent of any other disjoint subset of variables given a separating subset of variables:

$$\{ \langle \boldsymbol{X}_A \perp\!\!\!\perp \boldsymbol{X}_B \mid \boldsymbol{X}_C \rangle : \operatorname{sep}_{\mathcal{G}}(\boldsymbol{X}_A, \boldsymbol{X}_B \mid \boldsymbol{X}_C) \}. \tag{3.6}$$

These assumptions provide increasing guarantees, but are equivalent in case of a strictly positive distribution $\forall \boldsymbol{x} \colon P(\boldsymbol{X} = \boldsymbol{x}) > 0$ [106].

Factorization

According to the Hammersly-Clifford theorem [87, 82, 106], a positive probability distribution satisfies the Markov properties Eqs. (3.4) to (3.6) if and only if it is a "Gibbsian ensemble", i. e., its density factorizes over the *cliques* (fully connected / complete subgraphs) of the graph $\mathcal{G}$. Let $c \subseteq \mathcal{V}$ be a clique from a set of cliques $\mathcal{C}$ contained in $\mathcal{G}$ and $\phi_c \colon \prod_{i \in c} \mathcal{X}_i \to \mathbb{R}_{>0}$ a positive factor (also known as *potential function* or *clique potential*) of the clique c, the Gibbs distribution

$$P(\boldsymbol{X}) = \frac{1}{Z} \tilde{p}(\boldsymbol{x}) \tag{3.7}$$

can then be factorized over the cliques as unnormalized measure

$$\tilde{p}(\boldsymbol{x}) = \prod_{c \in \mathcal{C}} \phi_c(\boldsymbol{x}_c) \tag{3.8}$$

with

$$Z = \sum_{\boldsymbol{x} \in \mathcal{X}} \tilde{p}(\boldsymbol{x}) \tag{3.9}$$

being a normalization constant called the partition function. It is easy to see how the cliques c dictate the neighborhood in Eq. (3.5) and that a corresponding positive factorization fulfils those independence assumptions.

3.4.3 *Maximum a Posteriori Inference*

As we have seen, Markov random fields (MRFs) provide a principled and probabilistic way of describing potentially complex problems with many dependencies. A common mode of operation in this setting is to find the most probable explanation (MPE). This is generally referred to as maximum a posteriori (MAP) inference and corresponds to finding the state

$$\begin{aligned}\hat{\boldsymbol{x}} &= \arg\max_{\boldsymbol{x}\in\mathcal{X}} P(\boldsymbol{X}=\boldsymbol{x}) \\ &= \arg\max_{\boldsymbol{x}\in\mathcal{X}} \frac{1}{Z}\tilde{p}(\boldsymbol{x}) \\ &= \arg\max_{\boldsymbol{x}\in\mathcal{X}} \tilde{p}(\boldsymbol{x}) \\ &= \arg\max_{\boldsymbol{x}\in\mathcal{X}} \prod_{c\in\mathcal{C}} \phi_c(\boldsymbol{x}_c)\end{aligned} \tag{3.10}$$

that maximizes the joint probability.

Energy Representation

Often, a more convenient and numerically more stable formulation by taking the negative log of the clique potentials is used instead:

$$\psi_c(\boldsymbol{x}_c) = -\log \phi_c(\boldsymbol{x}_c) \tag{3.11}$$

In the following, we also refer to ψ_c as clique potentials (or potential functions) due to the one-to-one mapping to ϕ_c. This transformation allows to define the *energy* of the graphical model

$$E(\boldsymbol{x}) = \sum_{c\in\mathcal{C}} \psi_c(\boldsymbol{x}_c) \tag{3.12}$$

as a summation of the negative log transformations of the clique potentials, leading to the often—especially in statistical physics—seen formulation of the joint probability

$$P(\boldsymbol{X}) = \frac{1}{Z}\exp\left(-E(\boldsymbol{x})\right), \tag{3.13}$$

with the normalizing partition function Z. Due to the monotonicity of the negative log transformation, i. e., $P(\boldsymbol{X}) \propto -E(\boldsymbol{x})$, the MAP inference becomes finding the state

$$\begin{aligned}\hat{\boldsymbol{x}} &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} E(\boldsymbol{x}) \\ &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \sum_{c\in\mathcal{C}} \psi_c(\boldsymbol{x}_c)\end{aligned} \tag{3.14}$$

minimizing the MRF energy in Eq. (3.12) [117]. This can be viewed as the energy-based formulation to the original probabilistic one.

Inference Algorithms

Inference in discrete Markov networks is a combinatorial problem of energy minimization (Eq. (3.14)) considered $\mathcal{NP}$-hard in the worst case [106, p. 288]. The practical feasibility highly depends on the problem complexity and the actually used inference algorithm to compute Eq. (3.14).

The problem complexity is mainly dictated by the number of random variables N and the number of labels $\mathcal{X}_i$ for each one (i), as they describe the size of the state space $|\mathcal{X}| = \prod_{i \in \mathcal{V}} |\mathcal{X}_i| \leq (\max_{i \in \mathcal{V}} |\mathcal{X}_i|)^N$ that has to be searched. Additionally, the number and complexity of the used clique potentials adds to the overall search complexity as well. The actual inference runtime, however, highly depends on the chosen algorithm (i. e., exact versus approximate) and the graph and its parameterization. For example, *belief propagation*—also known as max-product—performs MAP inference in tree Markov networks in linear time. Note that there exist many other dedicated approaches exploiting special cases in computer vision [200]. For the general case however, see the comparative studies [96] and [101] evaluating various exact and approximate inference algorithms, respectively, on different computer vision problems.

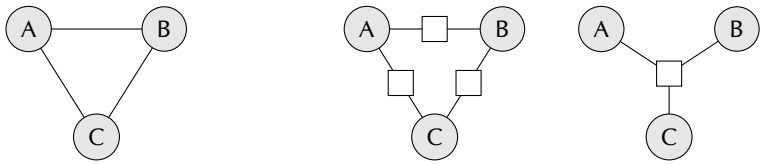
3.4.4 *Factor Graphs*

In order to explicitly describe the factorization of a Markov network (also of a Bayesian network), a *factor graph* [111] representation is often used. A factor node is introduced in order to explicitly describe the factorization of the joint distribution. This explicit description solves the problem of the original ambiguous graphical representation w. r. t. the factorization while encoding the same independence assumptions. See Fig. 3.11 for a graphical illustration of such ambiguity by showing two possibly induced factor graph representations for one MRF.

The unnormalized probability measure of a factor graph parameterized by a set of factors $\mathcal{F}$ where each factor $f \subseteq \mathcal{V}$ is encoded as a set of connected variable nodes, is simply given by the product of their potential functions:

$$\tilde{p}(\boldsymbol{x}) = \prod_{f \in \mathcal{F}} \phi_f(\boldsymbol{x}_f). \tag{3.15}$$

See the correspondence to the clique factorization in Eq. (3.8), given that a factor may encode a potential function over a clique. Thus, the



(a) Markov network (b) Two possibly induced factor graphs

Figure 3.11: The Markov network depicted in (a) can induce different factorizations, which are made explicit using the factor graph representations in (b). Note the additionally introduced factor node depicted by a square, allowing to explicitly encode the factorizations $\phi_1(A,B)\phi_2(B,C)\phi_3(A,C)$ and $\phi(A,B,C)$, respectively, in (b).

energy as in Eq. (3.12) and its inference Eq. (3.14) may be defined analogously for factor graphs.

Note that a Markov network with loops may have no loops in the corresponding factor graph representation (compare the second factor graph in Fig. 3.11b to the original graph representation in Fig. 3.11a). In this case, exact MAP inference in linear time is possible again [200].

3.4.5 *Conditional Random Fields*

A conditional random field (CRF) [113] uses the same graphical representation and parameterization as described in the previous subsection, but instead of encoding a joint distribution it encodes a *conditional joint distribution* $P(\boldsymbol{X} \mid \boldsymbol{D})$. Note the conditioning on some observed variables $\boldsymbol{D}$. This has a severe practical implication as it allows to incorporate observed variables whose dependencies might be very complex. Furthermore, it allows to define data-dependent potential functions without the need for a parametric model of the data. This is of great importance in computer vision as it allows to include all kinds of features without worrying of defining a correct joint distribution. [106, 200]

The independence assumptions of the Markov property follow analogously also for CRFs with a further conditioning on $\boldsymbol{D}$, and the—now conditional—Gibbs distribution is given by

$$P(\boldsymbol{X} \mid \boldsymbol{D}) = \frac{1}{Z(\boldsymbol{d})} \tilde{p}(\boldsymbol{x}; \boldsymbol{d}) \tag{3.16}$$

and the partition function

$$Z(\boldsymbol{d}) = \sum_{\boldsymbol{x} \in \mathcal{X}} \tilde{p}(\boldsymbol{x}; \boldsymbol{d}) \tag{3.17}$$

depends on the observed data $\boldsymbol{d}$.

Inference in context of CRFs corresponds to finding the solution that explains the observed data best using a discriminative model (rather

than a generative one). This is—especially in computer vision—a common mode of operation. Consider for example the task of generating a pixel-level segmentation [110] for an image or finding a human described by its joints [200, 65] in an image. Both tasks can be modeled using a CRF and solving the tasks corresponds to MAP inference:

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x} \in \mathcal{X}} P(\boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{D} = \boldsymbol{d}). \tag{3.18}$$

Part II
METHODOLOGY

4 KEY POINT DETECTION AND LOCALIZATION

The task is to *detect* and *localize*—if present—N different key points in a given digital image. For each of the N key points, it has to be decided whether this key point is contained in the given image and—if it is contained—the exact position of the key point has to be pinpointed. The task of labeling the individual key points is inherently solved in this setup. Note that detection is here defined as binary classification rather than the generation of target object bounding box as commonly done in computer vision.

The method described in the following solves both of those tasks, i. e., the detection and the localization, fully automatically and in a generic framework. However, it still allows to incorporate expert knowledge in the model building process, which can thus be tuned to specific tasks or applications. Furthermore, it is possible to easily incorporate export knowledge at test time in case of failure, operating the method in a semi-automatic fashion.

4.1 GENERAL IDEA

The method is comprised of two steps. First, *local appearance models* are used to find sets of likely positions—including alternatives—for the searched key points. Second, all combinations of likely positions are evaluated w. r. t. a *constellation model*, selecting the most likely one. This can be considered as going from local to global context, which is contrary to the common coarse-to-fine approach. The benefit here is that it is possible to use weak local appearance models that are easy to construct and quick to evaluate but might make mistakes. However, as long as one of the found alternative positions is correct, the second step has the chance to compensate for earlier mistakes.

Step 1: Generating Localization Hypotheses

More concretely, *local appearance models* are used to transform a given image $\boldsymbol{I}\colon \mathbb{N}^D \to \mathbb{R}$ into N key-point-specific pseudo (i. e., not normalized) probability maps $\boldsymbol{Y}_i\colon \mathbb{N}^D \to \mathbb{R}$ for $i = 1, \ldots, N$. Note that $D \in \{2, 3\}$ corresponds to the image dimensionality, either referring to planar or volumetric imaging, respectively. High values in these pseudo probability maps—commonly referred to as heatmaps—indicate the likelihood of the respective key point being located at the value's position. Thus, for each key point $i \in [1\,..\,N]$, non-maximum suppression (NMS)

is used to find the n_i strongest local maxima in the heatmap $\boldsymbol{Y}_i$. The positions of the local maxima represent the localization hypotheses $\mathcal{X}_i = \{\boldsymbol{x}_{i,1}, \dots, \boldsymbol{x}_{i,n_i}\}$, i. e., likely positions of the i-th key point. Note that non-correlation-exploiting methods would stop at this stage and select the position of the global maximum, i. e., the first best position $\boldsymbol{x}_{i,1} = \arg\max_{\boldsymbol{x}} \boldsymbol{Y}_i(\boldsymbol{x})$. But this position might not be the correct one. For instance, the prediction might be too far away from the true position or it might be confused with another ambiguous object. Additionally, the searched object might not be contained in the image at all. In order to solve those problems, a second correlation-exploiting step is employed.

Step 2: Finding a Joint Configuration

Given the N sets of localization hypotheses $\mathcal{X}_i$, a *constellation model* is used to select either one or none localization hypothesis for each key point $i \in [1\,..\,N]$. The respective selection is given by $s_i \in \mathcal{S}_i$, where $s_i \geq 1$ corresponds to the selection of the s_i-th localization hypothesis $\boldsymbol{x}_{i,s_i}$ while $s_i = 0$ corresponds to the selection of no localization hypothesis. Thus, the space of possible selections for the i-th key point is given by $\mathcal{S}_i = [0\,..\,n_i]$. Since $s_i = 0$ indicates the absence of a key point, it is referred to as the "missing" label. Note that the choice between the "missing" label and a localization hypothesis corresponds to the detection task, while the choice within the different localization hypotheses corresponds to the localization task. To this end, the constellation model evaluates each possible configuration $\boldsymbol{s} = \{s_i\}_{i \in [1..N]}$, i. e., the joint selection over all key points, from the set of all possible configurations $\mathcal{S} = \mathcal{S}_1 \times \dots \times \mathcal{S}_N$ and selects the most reasonable one $\hat{\boldsymbol{s}}$. This two-step process is visually depicted in Fig. 4.1.

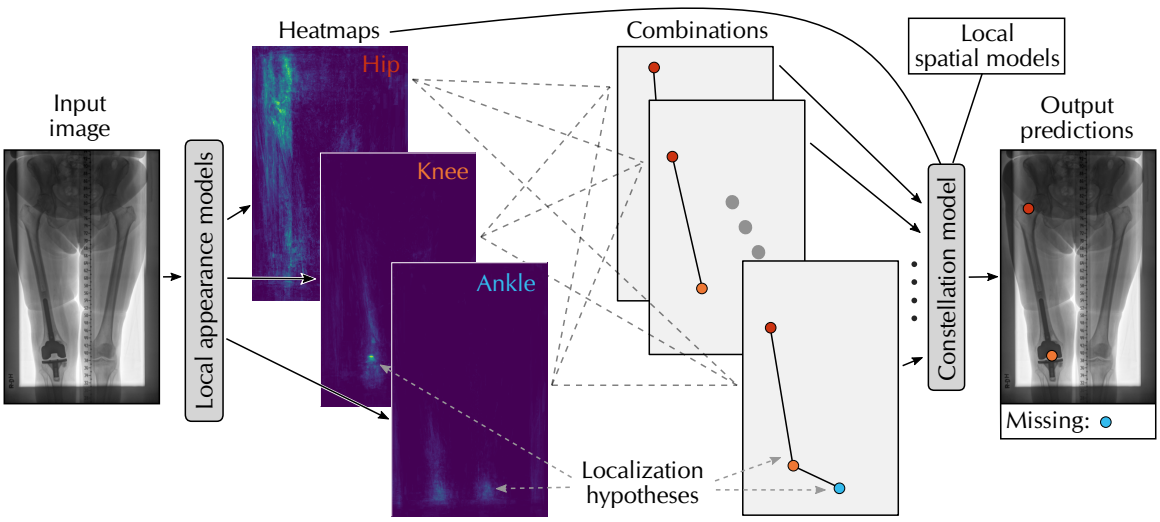


Figure 4.1: Illustration of the two-step detection and localization method using the example task of detecting and localizing the ■ hip, ■ knee and ■ ankle of the right leg (left in the image) in radiographs. First, local appearance models are used to generate key-point-specific heatmaps from which localization hypotheses are generated (local maxima; colored circles in the figure in the middle). Second, all possible combinations of localization hypotheses are evaluated w. r. t. the constellation model (consisting, e. g., of local spatial models and the heatmaps) to decide which key points are missing and pinpoint the positions of the detected key points (right side).

4.2 FORMULATION AS A CRF

The such specified detection and localization method utilizing local appearance models in combination with a constellation model can be formulated as a conditional random field (CRF; see Section 3.4.5) describing a joint probability distribution by establishing the following correspondences:

RANDOM VARIABLES The N key points correspond to N discrete random variables $\boldsymbol{S} = \{S_i\}_{i \in [1..N]}$, which are represented by nodes in the graphical model. Consequently, the random variable S_i takes on the values $\mathcal{S}_i$, where $0 \in \mathcal{S}_i$ corresponds to the "missing" label.

CRF STATE The configuration $\boldsymbol{s}$, i. e., the joint selection over all key points, corresponds to the joint assignment of values to the random variables $\boldsymbol{S}$ in the CRF. Consequently, the state space of the CRF is given by $\mathcal{S}$.

DATA CONDITIONING Since we are interested in a discriminative model w. r. t. the target image $\boldsymbol{I}$, the distribution is conditioned on that given image.

The such defined CRF is further parameterized by a *pool of F potential functions* $\Phi = \{\phi_f\}_{f \in [1..F]}$. These potential functions correspond to *knowledge sources* that evaluate the assumed positions of one up to theoretically N key points given the image $\boldsymbol{I}$. For instance, unary potential functions using just one key point position are derived from the

maxima in the heatmap $\boldsymbol{Y}_i$, while potential functions using more than one key point position are used to model the spatial constellation of key points.

However, it is often not possible to compute spatial information and thus potential functions for configurations involving the "missing" label $s_i = 0$. Thus, the MRF's energy from Eq. (3.12) is extended by dedicated value substitutes $\boldsymbol{B} = \{\beta_f\}_{f \in [1..F]}$—also referred to as "missing" energies—that are used in this case. Additionally, the fitness of each potential function is described by introducing potential-specific weights $\boldsymbol{\Lambda} = \{\lambda_f\}_{f \in [1..F]}$. We will later see (Chapter 7) how these weights are used to adapt the graph topology towards the target dataset and task. Finally, the CRF's energy (cf. Section 3.4.3)

$$E(\boldsymbol{s} \mid \boldsymbol{I}) = \sum_{f=1}^{F} \lambda_f \begin{cases} \beta_f, & \text{if } \exists i \in c_f \colon s_i = 0 \\ \psi_f\big(\boldsymbol{x}'(c_{f,1}), \dots, \boldsymbol{x}'(c_{f,C_f}); \boldsymbol{I}\big), & \text{otherwise,} \end{cases} \tag{4.1}$$

of the distribution $P(\boldsymbol{S} \mid \boldsymbol{I})$ is given by the summation of either the weighted negative log transformed potential function $\psi_f = -\log \phi_f$ (energy domain) or the respective weighted "missing" energy in case any key point in the clique $c_f \subseteq [1\,..\,N]$ is assigned the "missing" label $s_i = 0$, computed over F potential functions Φ. Note that each potential function gets only the selected positions $\boldsymbol{x}'(i) = \boldsymbol{x}_{i,s_i}$ from the sets of localization hypotheses of the key points that are contained in the clique $c_f = \{c_{f,j}\}_{j \in [1..C_f]}$ associated with the f-th potential function, which has an arity of C_f. Beware that multiple potential functions might be associated with the same clique (hence F might be larger than the number of cliques; compare the factor graph representation in Section 3.4.4), which are in practice combined (via a weighted summation) into one clique-specific potential function prior to any CRF inference.

Note that the "missing" energy β is the surrogate of an intrinsic property of the potential function while the weighting factor λ relates the potential function and thus its intrinsic properties to all other potential functions. This is why the "missing" energy as well as the potential function are both weighted. Consider for example a potential function of a clique of two nodes—commonly referred to as binary or pairwise potential function—using a Gaussian distribution to model the distance between both key points, which is qualitatively illustrated in Fig. 4.2 in a probabilistic fashion. One can see that the range of distances masked by the proxy β remains the same when changing the weight-

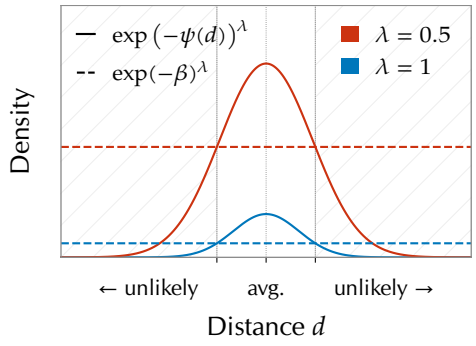


Figure 4.2: Probabilistic illustration of the relation between a distance-evaluating binary potential function $\psi(d)$ utilizing Gaussian statistics (solid lines) and the corresponding "missing" energy β (dashed lines) for two different values of λ (weightings).

ing λ. Furthermore it is evident that the weighting corresponds to a change of the spread when using Gaussian statistics. The intuitive notion of the "missing" label is the indication that the potential itself is uncertain as the underlying feature seems unlikely, which can be interpreted as one of the key points being absent due to a lack of a good localization.

4.3 JOINT DETECTION AND LOCALIZATION

Given such a defined CRF, the task of detecting and localizing the N key points boils down to a single MAP inference in graphical models, which is analogous to the minimization of the energy Eq. (4.1) and finding the optimal configuration

$$\hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s} \in S} E(\boldsymbol{s} \mid \boldsymbol{I}) \tag{4.2}$$

in terms of selected localization hypotheses and "missing" labels (analogous to Eq. (3.14)). Due to the formulation as a CRF, various inference algorithms with different properties can be applied in order to efficiently solve Eq. (4.2).

If not stated otherwise, A* search utilizing the admissible heuristic employing a tree-based lower bound estimate by Bergtholdt et al. [16, 5, 101] is used. It is—in theory—applicable to Markov networks of arbitrary topology and potential arity, while finding the exact first best configurations fast enough in this setup. However, any other applicable inference algorithm might be used, potentially exploiting the graph topology for further inference time improvements such as belief propagation in case of a tree-structured graph.

4.4 OVERCOMING INSUFFICIENT LOCALIZATION HYPOTHESES

The theoretical performance by the CRF in terms of localization is capped by an artificial upper bound, which is introduced by the reduction of the image domain to a set of likely localization hypotheses for each key point. This is not a problem if each set contains a hypothesis considered correct, but can be a problem if this is not the case. Given that the CRF always selects a prediction, an insufficient set and thus the forced selection of an incorrect localization hypothesis might negatively influence the selection of the remaining key points.

In case of a target dataset where the detection task is not necessary (i. e., it is a priori known which target key points are present in a given image), the semantic meaning of the "missing" label can be changed to "refine" in order to identify the key points for which the corresponding set of localization hypotheses is insufficient; still, the value $s_i = 0$ is kept to indicate this label. More concretely, after finding the optimal

selection $\hat{\boldsymbol{s}}$ using CRF inference (Eq. (4.2)), all key points that have the "refine" label ($s_i = 0$) assigned instead of a localization hypothesis ($s_i \geq 1$) are reevaluated. In order to assign those key points a position as well, all key points $\{S_i \mid \hat{s}_i \geq 1\}$ with a properly selected localization hypothesis $\boldsymbol{x}_{i,s_i}$ are fixed first. Then, each "refine"-labeled key point $\{S_i \mid \hat{s}_i = 0\}$ is individually optimized by considering all connected non-unary potential functions

$$\Psi_i = \left\{ \psi_f \in \Psi \mid i \in c_f \wedge C_f > 1 \wedge |\{i' \in c_f \mid \hat{s}_{i'} = 0\}| = 1 \right\} \tag{4.3}$$

that are fully specified—except for the current key point i—from the set of all negative log transformed potential functions $\Psi = \{\psi_f\}_{f \in [1..F]}$. The unary potential functions are excluded as we already know that they do not provide sufficient useful information, hence the necessary refinement. Given that this second inference

$$\tilde{\boldsymbol{x}}_i = \operatorname*{arg\,min}_{\boldsymbol{x} \in \boldsymbol{I}} \sum_{\psi'_f \in \Psi_i} \lambda_f \cdot \psi'_f(\boldsymbol{x}) \tag{4.4}$$

is performed on a very small subgraph, the search space can be extended to all discrete grid positions in the image $\boldsymbol{x} \in \boldsymbol{I}$ from just the set of localization hypotheses $\mathcal{X}_i$ ($|\mathcal{X}_i| \ll |\boldsymbol{I}|$), which would be intractable on the full graph. Note that some handwavy notation ψ'_f was used to indicate that the potential functions are computed by solely altering the position of key point i, since all others are known and fixed. By optimizing all "refine"-labeled key points in decreasing order of the number of connected potentials $|\Psi_i|$, previously refined positions can be used in the next optimization in terms of more usable potential functions. This also prevents the case that a key point may not have any usable potential function.

In case the target dataset poses the detection problem, the semantic meaning of the "missing" label ($s_i = 0$) can be changed to "unknown", as it is not clear whether not choosing a localization hypotheses was caused by a set of bad localization hypotheses or by the absence of the key point from the image. The refinement can still be carried out as before, but followed by a decision

$$\min_{\boldsymbol{x} \in \boldsymbol{I}} \sum_{\psi'_f \in \Psi_i} \lambda_f \cdot \psi'_f(\boldsymbol{x}) < \sum_{\psi'_f \in \Psi_i} \lambda_f \cdot \beta_f \tag{4.5}$$

whether the refinement was successful or not based on the associated "missing" energies.

4.5 IMPORTANT PARAMETERS

The complexity of the search problem formulated in Eq. (4.2) is mainly dictated by the number of key points N and the number of respective

localization hypotheses n_i, since both describe the number of configurations $|\boldsymbol{S}| = \prod_{i=1}^{N}(n_i + 1)$ contained in the search space $\boldsymbol{S}$. For large N or n_i this problem is infeasible; remember that inference in Markov networks, even approximate, is an $\mathcal{NP}$-hard problem [106].

4.5.1 *Localization Hypotheses*

While N is statically defined by the task at hand, i. e., the localization of a fixed set of key points, n_i is related to the image domain which can get arbitrarily large. Thus, one way to render the inference task feasible is to reduce n_i, which is done by considering only likely positions ($n_i = |\mathcal{X}_i|$) in the image $\boldsymbol{I}$ as evaluated by local appearance models. The number of *required localization hypotheses* is directly linked to the quality of the local appearance model. I. e., a very good model might have a correct localization hypothesis most of the time very early in the list of localization hypotheses, thus generally requiring a fewer amount. In contrast, a bad one might need a larger set as it might place a correct localization hypotheses later in the list. We describe two different heatmap-regressing local appearance models in Chapter 5, that are used to derive sets of localization hypotheses.

The set of likely positions might be constructed in a different fashion. For instance, one might use prior knowledge if the imaging setup follows a static regime and target structures are placed w. r. t. to a prior distribution. Alternatively, generative patterns might be used to create localization hypotheses without considering the image or prior information at all. Even another approach is to use very fast (compared to the local appearance) pre-filtering methods to remove unlikely localization hypotheses from the image domain. Such a setup might again be considered a coarse-to-fine approach as used in other approaches. Using such alternative schemes might provide the benefit of not having to evaluate the local appearance models over the whole image, but rather only a subset of likely positions, just constructed in a different kind of setup.

Additionally, it allows to operate the method in a semi-automatic fashion in contrast to fully automatic, which is especially useful in case of failure. For example, a radiologist might inspect the resulting predictions after a first CRF inference and spot a mis-localized key point, which eventually caused the mis-localization or inaccurate localization of other key points. Instead of manually fixing all predictions, the radiologist might provide a correct annotation $\hat{\boldsymbol{x}}_i$ for only one mis-localized key point and rerun the CRF inference while limiting the set of localization hypotheses for that key point to the annotated position $\mathcal{X}_i = \{\hat{\boldsymbol{x}}_i\}$. This might already fix all incorrect predictions, but can be repeated iteratively until all problems are solved. This is especially useful when working with many locally ambiguous key points in combination with

restricted field of views, which tend to provide accurate predictions but incorrect labels (shifts).

4.5.2 *Potential Functions as Knowledge Sources*

Another important parameter are the types of potential functions used, as they can be considered the *knowledge sources* of the CRF by evaluating a given configuration formed by the combination of localization hypotheses. The design of potential functions is basically a feature engineering effort, since most potential functions derive some sort of feature and evaluate the likelihood of the feature w. r. t. to the detection and localization task. Note that a potential function depends on a fixed number of key points to operate, which is referred to as the arity of the potential function (e. g., unary, binary, ternary, etc.). While potential functions with a higher arity generally allow to infer richer features, they also render the inference problem considerably harder.

One very obvious source of knowledge are the used local appearance models, as they generally provide information about the key point positions since they are used to derive likely positions. Thus, their heatmaps are represented by unary potential functions in the CRF. The local appearance models are later introduced in Chapter 5.

Potential functions of higher arity (i. e., larger than one) are used to evaluate the constellation of small groups of key points, as they evaluate multiple key point positions. Usually, they have a scope of only 2 or 3 key points. This facilitates the factorization of the CRF and reduces the estimation of a global constellation model to a combination of local potential functions evaluating small key point cliques (this resonates the "product of experts" [93] idea). Such *clique constellation models* are later described in Chapter 6.

Note that the combination of unary and non-unary potential functions is necessary in order to appropriately handle cases with missing key points not inflicted by a reduced field of view. Consider for example missing limbs (an example is the second to last image in Fig. 10.4) in combination with uncut field of view, a non-unary spatial model would yield a strong response in place of the standard position and hence needs additional information to down-weight that influence, which is not the case for the problem of a missing key point inflicted by a reduced field of view.

4.5.3 *Graph Topology and "Missing" Energies*

A central problem is the definition of the graph's topology, which corresponds to the construction of a factor graph (see Section 3.4.4) by adding potential functions for specific key point cliques. Effectively,

this establishes the dependencies between key points based on some potential specific features. One extreme is the usage of a "fully loaded" graph as in [16], i. e., using all potential functions for all possible arity-matching cliques. However, this might include unnecessary and potentially performance degrading information. On the opposite end, heuristic [60, 58] or prior-knowledge-based [172, 78] definitions might miss important connections.

To this end, an energy-based optimization method starting from a *pool of potential functions* to automatically select the most relevant potential functions by optimizing the potential weights $\boldsymbol{\Lambda}$ and therefore defining the factor graph is proposed in Chapter 7. Additionally, it optimizes the "missing" energies $\boldsymbol{B}$ without resorting to heuristic approaches as in [16].

5 LOCAL APPEARANCE MODELS

As described earlier, *local appearance models* are represented by unary potential functions in the CRF (similar to, e. g., [16, 59]) and their task is two-fold. First, they evaluate the image appearance $\boldsymbol{I}$ and compute a heatmap $\boldsymbol{Y}_i$ for each key point i. These heatmaps are in turn used to derive the key-point-specific unary potential functions

$$\phi_i(\boldsymbol{x}_i\,;\,\boldsymbol{I}) = \boldsymbol{Y}_i(\boldsymbol{x}_i). \tag{5.1}$$

Note that these functions are only defined for $s_i > 0$, which is ensured by the filter in Eq. (4.1). Second, for each key point i, localization hypotheses $\mathcal{X}_i$ are generated by applying non-maximum suppression (NMS) to the corresponding heatmap $\boldsymbol{Y}_i$ in order to render the CRF inference task feasible. Note that it is crucial to not exclude the correct position from the set of localization hypotheses, which can be facilitated by the usage of *local context* only. On the one hand, this potentially decreases the localization performance of the model itself (global maximum), but on the other hand it should increase the quality of the many local maxima. I. e., the first local maxima might not be correct, but a correct one should follow quickly after that, which is a necessary constraint for the CRF to find a correct solution.

Note that the two tasks, i. e., evaluating the local image appearance and reducing the set of possible key point positions, can also be replaced by simple "ad hoc" methods. E. g., in case of a tree-structured factor graph, it might not be necessary to reduce the label set at all. In contrast, in fixed rigid setups prior information might be used to propose likely positions without evaluating the local appearance models at every possible position or another much faster coarse model for pre-filtering [59] might be used. Furthermore, multiple appearance models for each key point might be used to improve the robustness. However, in the following, two local appearance models are described to support the general case and thus solve both tasks efficiently.

5.1 REGRESSION TREE ENSEMBLES

The first type of appearance model uses ensembles of T_i decision trees—one ensemble for each key point i—to regress a pseudo probability $y_i(\boldsymbol{x})$ from a feature vector computed for a small patch around the position $\boldsymbol{x}$.

5.1.1 *Feature Extraction*

A binary robust independent elementary features (BRIEF)-like approach [34] is used to compute V_i intensity difference features for values randomly sampled within a small patch $\boldsymbol{A}_i$ centered around the position $\boldsymbol{x}$, thus generating a sparse representation of $\boldsymbol{A}_i$. The values are sampled by using tree-specific offset vectors, which are initially generated for each tree in order to improve the generalization of the ensemble [25] using a Gaussian distribution. Such offset vectors computed for a certain patch size are visualized in Fig. 5.1. The patch size $\boldsymbol{A}_i$ relates to the target object's size—the knee in this example—and is used to construct the Gaussian distribution $\mathcal{N}(\mathbf{0}, \operatorname{diag}(1/5\boldsymbol{A}_i)^{\circ 2})$ used for the initial sampling of offset vectors ($\boldsymbol{X}^{\circ 2}$ indicates the Hadamard square / element-wise square of $\boldsymbol{X}$). The factor $1/5$ is chosen such that 2.5 standard deviations σ correspond to half the patch size, ensuring that the object is better represented within a close proximity to $\boldsymbol{x}$ while less so towards the patch's border (compare Fig. 5.1). Note that the first half of the V_i features computes the difference between the value at $\boldsymbol{x}$ and a random offset, while the second half uses two offsets to compute the difference. In case of multi-channel images, the total amount of features is evenly distributed over the different channels.

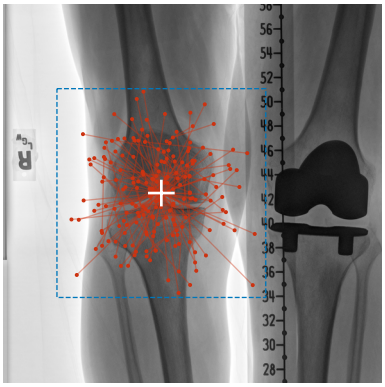


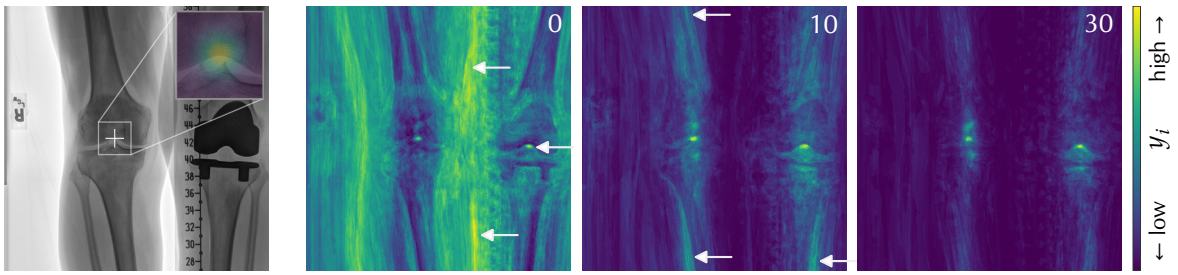
Figure 5.1: Example illustration of the intensity value differences (■ dots connected by lines) computed for a certain position $\boldsymbol{x}$ ("+") within a certain patch size $\boldsymbol{A}_i$ (■ rectangle) around the target object knee.

In contrast to the original BRIEF approach, the binarization is dropped as this can be achieved by the decision tree using a zero-threshold and should be learned automatically if this is a desirable feature. Although the features are already quick to compute, it is not necessary to compute all V_i features at inference, given that only one feature value is evaluated at each tree node and that the tree depth is usually much smaller than V_i. Furthermore, tree inference can be implemented to utilize the GPU in order to further speed up the processing [178]. Note that these features and thus the tree-based regression work for 2D as well 3D images without loss of generality.

From now one, we refer to this method as regression tree ensembles (RTEs) rather than random forest (RF), since the only source of randomness is the tree-specific sampling mask, which has been shown [34] to provide a very discriminative subsampling.

5.1.2 *Iterative Discriminative Training*

A central problem when constructing / training the regression trees is the selection of representative feature vectors (training samples). While



(a) Test image (b) Corresponding heatmap over the course of the discriminative training

Figure 5.2: Illustration of the (b) regression tree ensemble's heatmap output over the course of the discriminative training generated for a (a) random test image when targeting the right knee (white "+"). The normal distribution used as regression target is depicted as color-coded overlay in the inset in (a). The images in (b) show the ensemble's output after evaluating feedback of (from left to right) 0, 10 and (total) 30 training images (note that the ensembles of intermediate trees are generated only for illustration purposes and are not used during training); strong incorrect responses have been annotated with white arrows. Note the additional peak on the incorrect left knee which is a desired feature of such local appearance models as the ambiguitiy is solved by the constellation model.

it is quite reasonable to use feature vectors computed for positions close to the annotated key point's position as "positive" training samples (i. e., regression target value greater than 0), it is not clear which feature vectors should be used as "negative" training samples (regression target value of 0). Using all non-close positions as negative samples creates a high class imbalance and includes potentially highly correlated feature vectors, while a random selection might miss important discriminating feature vectors. Thus, an iterative discriminative feature vector selection approach based on intermediate decision trees is proposed.

The following process is done for each tree in the ensemble independently (in contrast to, e. g., boosting approaches [70]); remember that each tree is associated with a unique and tree-specific sampling mask in order to generate the feature vectors (training samples). Central idea is—for each member of the ensemble—to train a sequence of intermediate regression trees, based on a fixed set of positive training samples and a variable and growing set of negative samples. The intermediate trees are only used to generate high-quality negative samples from input images, which do not belong to the set of images on which the intermediate trees were trained on. This process is explained in more detail below. When all training images have been used to generate negative training samples, a final tree is trained and added to the ensemble, while the intermediate trees are discarded.

Generation of Positive Samples:

The set of positive samples (which remains fixed for each intermediate tree as well as for each member of the ensemble) is collected from each training image by computing the feature vectors for all positions $\boldsymbol{x}$ whose Euclidean distance to the respective annotated position $\dot{\boldsymbol{x}}$ is

smaller than some threshold R, i. e., $\|\boldsymbol{x} - \dot{\boldsymbol{x}}\|_2 < R$. The corresponding target regression value for each feature vector is generated by placing a normal distribution $\mathcal{N}(\dot{\boldsymbol{x}}, (1/3R\boldsymbol{I}_D)^{\circ 2})$ at the annotated position $\dot{\boldsymbol{x}}$ such that $\sim 99\,\%$ of the probability mass is within radius R ($3\sigma = R$) and computing the density value at $\boldsymbol{x}$ (see inset in Fig. 5.2a). This provides high values close to the annotated position and very low values outside a small neighborhood R. For each ensemble member, a first intermediate regression tree is constructed on this set of "positive" training samples (again, each member of the tree uses a different sampling mask), using variance reduction in combination with a low maximal tree depth (commonly 15) to ensure generalization (see Section 3.2.1). Since these trees have not seen "negative" samples in training to discriminate to, their output is rather unspecific. This is illustrated as first image in Fig. 5.2b, where all first intermediate trees are combined to an ensemble, which is applied to the test image (a) and averaged the output of the individual trees. Note that the ensemble is generated only for illustration purposes; it is not used in the training procedure.

Incremental generation of negative training samples:

The key idea now is to generate highly discriminative negative training samples, which are added to the training samples of each intermediate tree. To improve generalization, those negative training samples should be generated on images that have not been employed in training of the intermediate tree (except for generating positive samples). On the other hand, the intermediate tree should be powerful enough to generate highly discriminative negative training samples. To achieve both goals, a sequence of intermediate regression trees is trained in an iterative, incremental process, trained on the fixed set of positive samples and a set of negative samples which is increased in each iteration, and then applied to training images not included (for negative training samples) in the training process so far. More precisely, the set of training images is divided into a set of B mini-batches. Then, each first intermediate tree (of each ensemble member) is applied to the first mini-batch of input images, generating heatmaps from which "negative" training samples are extracted. More specifically, NMS is used to find the M strongest local maxima in each image of the mini-batch not within radius R to the annotated position $\dot{\boldsymbol{x}}$, where M is the number of positive training samples extracted per training image (to ensure class balance). For these M positions per training image (and member of the ensemble), the corresponding feature vectors are computed (using the respective sampling mask) and added to the current set of training samples with a regression target value of 0. On this enlarged set of training samples (keeping the set of positive samples fixed), a new intermediate regression tree is trained as described above. This new intermediate tree is used to generate negative training samples in the next iteration, which are again added to the current training set. After processing

the last mini-batch (i. e., all training images), the last intermediate tree (now trained on the full set of positive and full set of negative training samples) is used as final tree and added to the ensemble, while the previous intermediate trees are discarded. How the ensemble's response changes over the course of this discriminative training setup is illustrated in Fig. 5.2b.

5.1.3 *Important Parameters*

While there are theoretically many parameters to tune, it is only necessary to adjust a subset of those in order to target different anatomical objects while obtaining decent performance.

PATCH SIZE The patch size $\boldsymbol{A}_i$ specifies the area that is seen when computing feature vector offsets and thus creating the sparse representation of a patch. Given that this is the only information the trees are using in order to regress a heatmap value for a respective location, it should be ensured that the size is as large as needed to cover the important object structure while being as small as possible to focus on local context only. This parameter can easily be obtained by assessing the dataset and object at hand.

NUMBER OF FEATURES As previously mentioned, the number of computed features per patch V_i ensures sparsity, thus it should be much smaller than the number of pixels / voxels in the patch. Furthermore, it should generally be larger for 3D images than for 2D images due to the added dimension. Empirically, values in the range of 128 to 512 have shown to generally provide good results.

TREE DEPTH The tree depth influences the generalization capability to unseen data and should generally be much smaller than the amount of available features V_i. Empirically, values in the range of 10 to 20 have shown to generally provide good results.

NUMBER OF TREES The ensemble size T_i (i. e., the number of trees) directly influences the resulting localization performance. Remember that each tree uses a different randomly generated feature sampling mask, which greatly improves the generalization capability the more trees are being used. While already very few trees in an ensemble create localization hypotheses $\mathcal{X}_i$ containing at least one "correct" hypothesis, the position of it, which is influenced by the strength of the response, can be improved by simply increasing the amount of used trees. This allows to adjust the trade of improved localization performance for degraded runtime performance by altering T_i. Empirically, values in the range of 32 to 144 have shown to provide good results.

5.2 CONVOLUTIONAL NEURAL NETWORK

An alternative approach is the usage of CNNs—especially fully convolutional networks—as local appearance models, since they have shown superior results in a lot of areas in computer vision. In contrast to the previously described method, they drop the need to manually engineer the features in favor of composing a suitable CNN architecture. Luckily, there already exists a plethora of readily available architectures to choose from.

5.2.1 *U-Net Architecture*

A very robust and versatile architecture is the fully convolutional [124] U-Net architecture proposed by Ronneberger et al. [164]. It has shown superior results in various medical imaging tasks including medical segmentation challenges. Furthermore, it has been shown [98] that the U-Net even outperforms various other task-specific networks just by using a proper training regime, diminishing the need for yet another CNN architecture.

The U-Net architecture is comprised of an encoding path and a decoding path, which both are connected at different levels via skip connections. The encoding path step-wise doubles the receptive field size by reducing the spatial resolution and at the same time doubling the number of feature maps. Thus, it builds an increasing abstract feature representation enforced by a bottleneck right before upsampling again. The decoding path reverses this procedure, but is enriched by feature maps taken from the same levels of the encoding path. The increase in spatial resolution is achieved by a 2×2 transposed convolution of stride 2, which halves the feature maps while doubling the output size. In the end, a 1×1 convolution (also known as network in network [121]) reduces the feature maps to 2 channels and finally applies a soft-max activation to arrive at the binary segmentation. See [164] for a more thorough description.

5.2.2 *Modifications*

In order to use the U-Net for key point localization, only few things have to be changed. Again, we consider regressing pixel-wise pseudo probabilities forming key-point-specific heatmaps $\boldsymbol{Y}_i^{\text{unet}}$ as solving the localization task. To do so, we set the number of feature maps generated by the 1×1 convolution to the number of key points N and drop the final soft-max activation function in order to facilitate independent regression of target values. Furthermore, it is assumed that the hierarchical feature representation is beneficial in modeling the appearance of multiple objects due to common features especially with locally ambiguous

objects. Thus, exactly one network is used to regress the N heatmaps for all key points at the same time. To ease the engineering effort, feature maps are zero-padded prior to the 3×3 convolutions in order to obtain full-size heatmaps without the need for a tiling strategy when using images of size $X \times Y$ as input, nor requiring to crop the feature maps of the skip connections prior to concatenation. Note that the image size has to be a multiple of 16 due to the discretized downsampling, which is practically ensured by zero-padding the image (if necessary) and cropping the resulting heatmap. The modified U-Net architecture w. r. t. the localization task is depicted in Fig. 5.3.

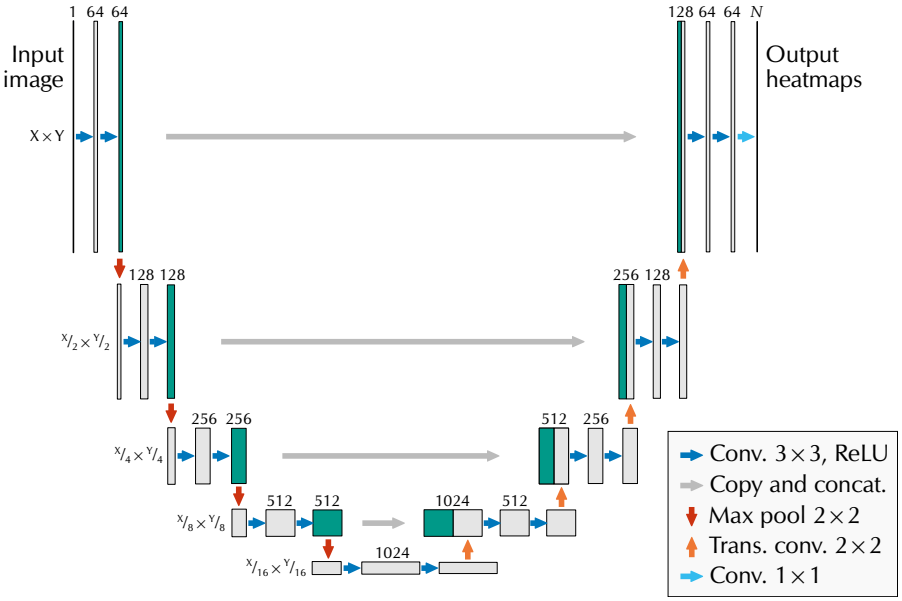


Figure 5.3: Illustration of the modified U-Net CNN architecture used for the regression of N heatmaps for 2D images of size $X \times Y$. Note the skip connections from the encoding path (downsampling; left) to the decoding path (upsampling; right). In contrast to the original U-Net architecture, the output heatmaps have the same size as the input image. Based on [164, Fig. 1].

Similarly to the previous regression-tree-based method, the U-Net may be applied to 3D images by using the analogous 3D convolutional and pooling operations. However, this is often practically not possible due to the increased memory demand, longing for different architectures in combination with an image-tiling strategy. Possible well-known CNN candidates are the 3D U-Net [45], which mostly follows the same encoder-decoder pattern as the 2D U-Net, or the V-Net [139]. In this thesis however, we only use the (modified) 2D U-Net as we evaluate it only with 2D images. This is a general problem of CNN-based approaches that try to generalize from 2D to 3D, in contrast to the previously introduced regression tree ensembles.

5.2.3 *Training*

Similarly to the previous method, the density of a normal distribution $\mathcal{N}(\mathring{\boldsymbol{x}}, (1/3R\boldsymbol{I}_D)^{\circ 2})$ placed at the annotated position $\mathring{\boldsymbol{x}}$ (recall the inset in Fig. 5.2a) is used to construct the regression target feature map $\mathring{\boldsymbol{Y}}_i^{\text{unet}}$. Stochastic gradient descent (SGD) in form of the Adam algorithm [104] is used to optimize a sum of squared errors (SSE) loss function created between the predicted heatmap $\boldsymbol{Y}_i{}^{\text{unet}}$ and the target heatmap $\mathring{\boldsymbol{Y}}_i^{\text{unet}}$, allowing the network to learn the necessary features.

Given that the number of trainable parameters (i. e., convolutional weights and biases) is much larger than the amount of available training data, data augmentation has been advocated in [164]. Thus, elastic deformation as proposed in [182] is used to further add 10 times the training amount in form of augmented training images. The training is carried out for 1000 epochs using a mini-batch size of 8 patches per iteration.

5.2.4 *Important Parameters*

The black box nature of CNNs is somewhat present in the limited amount of intuitive analogies for available parameters to tune. The definition of the architecture can be considered an additional parameter, which provides an endless search space (the automatic creation of architectures is a research field in itself). Thus, it is reasonable to settle for a proven architecture as the U-Net. However, this also fixes the receptive field size to 200×200 px in case of the modified U-Net, which is the analogue to our previous patch size as it limits what the network sees when regressing the value for a certain pixel. As explained earlier, the receptive field size should be related to the target object size, effectively requiring dedicated target-object specific architectures or novel concepts like the OBELISK [90].

Apart from that, there are other important parameters that mainly adjust the training setting and ensure that proper feature representations are learned:

LEARNING RATE While Adam provides more tunable parameters than conventional SGD, the default settings proposed in [104] generally yield better results without requiring the need for additional methods to achieve similar performance. The main parameter in this case is the learning rate (global step size in [104]), which has shown to yield good results in the range of 1E−4 to 1E−5.

NUMBER OF EPOCHS The number of epochs or the number of iterations specifies how many weight updates are performed over the course of the optimization. Generally, the number should be large enough to reach convergence, but small enough to not overfit

on the training data. However, we have found that using large amounts of augmented training images reduces the effect of overfit quite dramatically and allows to run for large number of epochs with time being the major constraint.

TARGET VALUE The value range of the regression target should be a magnitude larger than the value range of the input image, as it not only enables a successful training, but it also accelerates it. This has been empirically verified for different CNN-based regression networks and different target datasets.

6 CLIQUE CONSTELLATION MODELS

In the following, potential functions of higher arity are discussed (recall Section 4.5). The evaluation of the key point constellation, i. e., of a state $\boldsymbol{s}$ of the CRF (see Eq. (4.1)), is performed by various individual non-unary potentials functions, i. e., knowledge sources that take not just one key point hypothesis into account but the hypotheses of multiple key points. In order to facilitate the CRF factorization, these potential functions only operate on a subset of key points and correspond to *clique constellation models*, while the joint formulation represents the actual constellation model.

6.1 BINARY SPATIAL STATISTICS

The most simple form of clique constellation models are potential functions that depend on the positions of two key points only, commonly referred to as pairwise or binary potential functions [60, 16, 59, 78, 41, 198, 4, 89]. While potential functions exist that evaluate image appearance features [60, 16, 41], they are not commonly used as they significantly increase the runtime. This is in contrast to potential functions that operate just on spatial information, which is generally easy to estimate and compute. In the following, three simple binary potential functions operating just on spatial features are described. They are computable for 2D as well as 3D images and provide different invariance properties. Note that all potential functions discussed in the following are (globally) translation invariant.

6.1.1 *Distance*

The first potential function computes the Euclidean *distance* between two key point positions and uses a normal (Gaussian) distribution to model it (Donner et al. [59] use multiple normal distributions with empirically chosen standard deviation and Bergtholdt et al. [16] use a histogram instead of a distribution). For two key points i and j, the potential function

$$\phi_{i,j}^{\text{dist}}(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \mu_{i,j}^{\text{dist}}, \sigma_{i,j}^{\text{dist}}) = f\left(d(\boldsymbol{x}_i, \boldsymbol{x}_j) \,\middle|\, \mu_{i,j}^{\text{dist}}, \sigma_{i,j}^{\text{dist}}\right) \tag{6.1}$$

computes the distance

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_2 \tag{6.2}$$

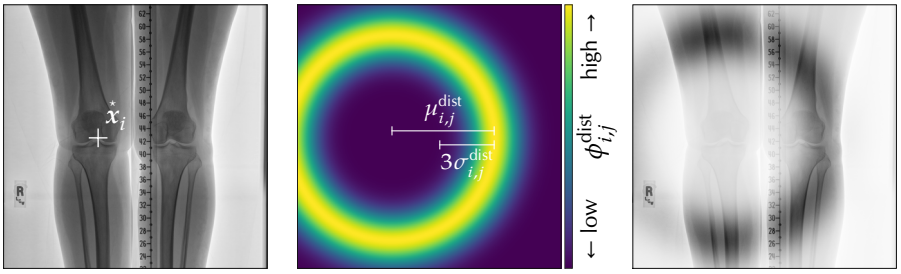
and uses the probability density function of the normal distribution

$$f(x \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \tag{6.3}$$

to compute the probability density for the key point distance. The normal distribution is parameterized by the mean distance $\mu_{i,j}^{\text{dist}}$ and its standard deviation $\sigma_{i,j}^{\text{dist}}$, both estimated by maximum likelihood on training data (see next subsection).

Note that this (and each following) potential function is only defined if $s_i > 0$ and $s_j > 0$, i. e., no "missing" label is selected. This is ensured by the filter used in the joint CRF formulation from Eq. (4.1). However, it is also possible to define potentials neither requiring this filter nor the estimation of the missing energy β, which allows to define potential functions evaluating combinations of missing labels and positions in order to, e. g., model anatomical appearance constraints.

The resulting pseudo probability of the potential function Eqs. (6.1) to (6.3) is qualitatively illustrated in Fig. 6.1 for a given target image by fixing one key point and varying the second key point over the target image. Note how the rotation invariance is visualized by the circular pattern in Fig. 6.1b.



(a) Target image (b) Potential function (c) Combination

Figure 6.1: Qualitative illustration of the distance-evaluating potential function Eqs. (6.1) to (6.3) by assuming that both key points are the knees and fixing one position to the annotated position and varying the other one. (a) A sample target image with the annotated position for the right knee $\dot{x}_i$ marked with a "+". (b) Result of evaluating the potential function when varying the position of the left knee over the image domain while fixing the right one. (c) Modulation of the target image's transparency with the potential values to get a better intuitive understanding. Hence, more transparent areas indicate a lower probability.

Training

The training of this distance-evaluating potential function Eqs. (6.1) to (6.3) corresponds to estimating the sufficient statistics $\mu_{i,j}^{\text{dist}}$ and $\sigma_{i,j}^{\text{dist}}$

of the normal distribution Eq. (6.3). Applying MLE based on a set of K annotated training key point positions $\overset{\star}{\boldsymbol{x}}_i$ and $\overset{\star}{\boldsymbol{x}}_j$, this amounts to simply computing [21, pp. 93–94] the data mean

$$\mu_{i,j}^{\text{dist}} = \frac{1}{K} \sum_{k=1}^{K} d(\overset{\star}{\boldsymbol{x}}_i^k, \overset{\star}{\boldsymbol{x}}_j^k) \tag{6.4}$$

and data standard deviation

$$\sigma_{i,j}^{\text{dist}} = \sqrt{\frac{1}{K-1} \sum_{k=1}^{K} \left(d(\overset{\star}{\boldsymbol{x}}_i^k, \overset{\star}{\boldsymbol{x}}_j^k) - \mu_{i,j}^{\text{dist}} \right)^2} \tag{6.5}$$

corrected for its biased expectation.

6.1.2 *Angle*

The second potential function computes the *angle* between the line passing through both key point positions and the first axis of a suitably chosen plane (if the input dimension D is larger than 2), and uses a circular normal (von Mises) distribution [137, p. 36] to model it (in [16], a histogram is used instead). For two key points i and j, the potential function

$$\phi_{i,j}^{\text{ang}}(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \mu_{i,j}^{\text{ang}}, \kappa_{i,j}^{\text{ang}}) = g\left(\alpha(\boldsymbol{x}_i, \boldsymbol{x}_j) \,\middle|\, \mu_{i,j}^{\text{ang}}, \kappa_{i,j}^{\text{ang}} \right) \tag{6.6}$$

first computes the angle

$$\alpha(\boldsymbol{x}_i, \boldsymbol{x}_j) = \operatorname{atan2}(v_2, v_1) \tag{6.7}$$

between the projected $\rho\colon \mathbb{R}^D \to \mathbb{R}^2$ difference vector $\boldsymbol{v} = (v_1\ v_2)^T = \rho(\boldsymbol{x}_j - \boldsymbol{x}_i)$ of the key point positions $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ and its first axes (see Fig. 6.2). Beware, $\operatorname{atan2}\colon \mathbb{R} \times \mathbb{R} \to (-\pi, \pi]$ is the extended arctangent supporting all four quadrants. Then, the probability density function of a circular normal distribution

$$g(x \mid \mu, \kappa) = \frac{1}{2\pi I_0(\kappa)} \exp\left(\kappa \cos(x - \mu)\right) \tag{6.8}$$

is used to compute a probability density for the angle given the mean angle $\mu_{i,j}^{\text{ang}}$ and its concentration $\kappa_{i,j}^{\text{ang}}$. Note that I_0 corresponds to the modified Bessel function. In case of 2D images, the projection ρ corresponds to the identity function while it has to be specified explicitly for 3D images, e. g., using the medical planes coronal, sagittal and transverse.

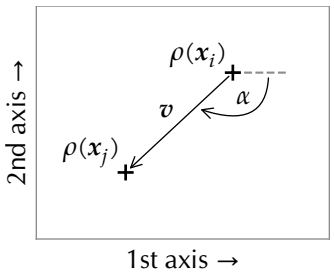
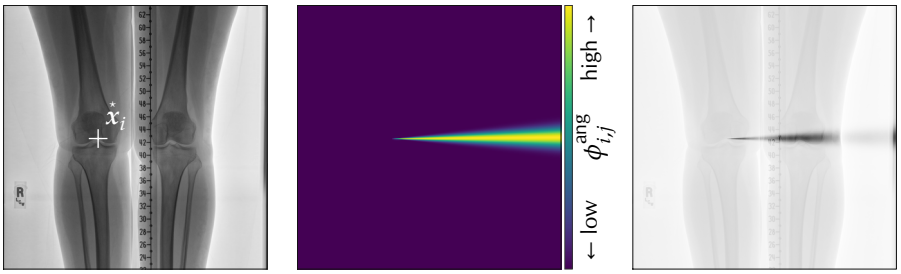


Figure 6.2: Illustration of the computed angle α for the projected (ρ) difference vector $\boldsymbol{v}$ w. r. t. both key point positions $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

The resulting pseudo probability of the potential function Eqs. (6.6) to (6.8) is qualitatively illustrated in Fig. 6.3 for a given target image by fixing one key point and varying the second key point over the target

image. Note how the scaling invariance is visualized by the infinite radius of the arc in Fig. 6.3b.



(a) Target image (b) Potential function (c) Combination

Figure 6.3: Qualitative illustration of the angle-evaluating potential function Eqs. (6.6) to (6.8) by assuming that both key points are the knees and fixing one position to the annotated position and varying the other one. (a) A sample target image with the annotated position for the right knee $\dot{x}_i$ marked with a "+". (b) Result of evaluating the potential function when varying the position of the left knee over the image domain while fixing the right one. (c) Modulation of the target image's transparency with the potential values to get a better intuitive understanding. Since both legs have roughly the same length, the concentration is very high.

Training

The training of the potential function Eqs. (6.6) to (6.8) corresponds to estimating the sufficient statistics of the von Mises distribution Eq. (6.8) for which, again, MLE is applied. For the mean orientation $\mu^{\text{ang}}_{i,j}$, this corresponds [137, p. 85] to computing the angle

$$\mu^{\text{ang}}_{i,j} = \operatorname{atan2}(r_2, r_1) \tag{6.9}$$

of the overall direction of all K training samples

$$\boldsymbol{r} = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \sum_{k=1}^{K} \begin{pmatrix} \sin \alpha(\dot{\boldsymbol{x}}_i^k, \dot{\boldsymbol{x}}_j^k) \\ \cos \alpha(\dot{\boldsymbol{x}}_i^k, \dot{\boldsymbol{x}}_j^k) \end{pmatrix}. \tag{6.10}$$

The computation of the concentration $\kappa^{\text{ang}}_{i,j}$ involves the ratio of modified Bessel functions, which allows for no analytic solution. Thus, we resort to the asymptotic approximation suggested in [10] to compute

$$\kappa^{\text{ang}}_{i,j} = \frac{2\bar{r} - \bar{r}^3}{1 - \bar{r}^2}, \tag{6.11}$$

with

$$\bar{r} = \frac{\|\boldsymbol{r}\|_2}{K} \tag{6.12}$$

being the Euclidean distance from the unit vector barycenter to the origin.

6.1.3 *Vector*

The third potential function computes the *vector* between both key point positions and uses a multivariate normal distribution to model it. For two key points i and j, the potential function

$$\phi_{i,j}^{\text{vec}}(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\mu}_{i,j}^{\text{vec}}, \boldsymbol{\Sigma}_{i,j}^{\text{vec}}) = h\left(v(\boldsymbol{x}_i, \boldsymbol{x}_j) \,\middle|\, \boldsymbol{\mu}_{i,j}^{\text{vec}}, \boldsymbol{\Sigma}_{i,j}^{\text{vec}}\right) \tag{6.13}$$

computes the vector

$$v(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_j - \boldsymbol{x}_i \tag{6.14}$$

and uses the probability density function of the multivariate normal distribution

$$h(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D \det(\boldsymbol{\Sigma})}} \exp\left(\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right) \tag{6.15}$$

to compute a pseudo probability for it given the mean vector $\boldsymbol{\mu}_{i,j}^{\text{vec}}$ and its covariance matrix $\boldsymbol{\Sigma}_{i,j}^{\text{vec}}$.

The resulting pseudo probability of the potential function Eqs. (6.13) to (6.15) is qualitatively illustrated in Fig. 6.4 for a given target image by fixing one key point and varying the second key point over the target image. From Fig. 6.4b one can see that this potential is neither scaling nor rotation invariant. It might be viewed as a combination of both former potential functions, since the vector encodes the rotation as well the distance.

Training

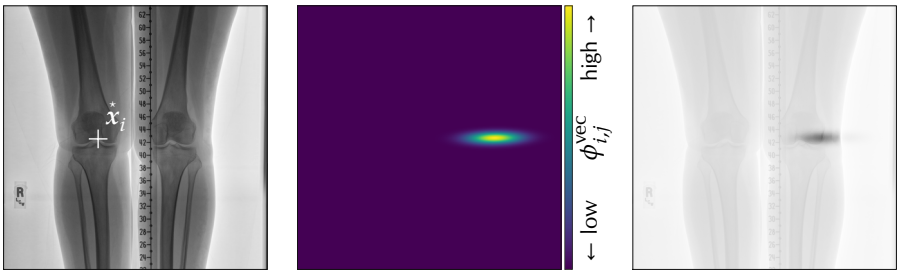
As before, MLE is applied to find the parameters of the multivariate normal distribution Eq. (6.15), which again corresponds [21, p. 93] to finding the data mean

$$\boldsymbol{\mu}_{i,j}^{\text{vec}} = \frac{1}{K} \sum_{k=1}^{K} v(\overset{\star}{\boldsymbol{x}}_i^k, \overset{\star}{\boldsymbol{x}}_j^k) \tag{6.16}$$

and the data covariance

$$\boldsymbol{\Sigma}_{i,j}^{\text{vec}} = \frac{1}{K-1} \sum_{k=1}^{K} \left(v(\overset{\star}{\boldsymbol{x}}_i^k, \overset{\star}{\boldsymbol{x}}_j^k) - \boldsymbol{\mu}_{i,j}^{\text{vec}}\right) \left(v(\overset{\star}{\boldsymbol{x}}_i^k, \overset{\star}{\boldsymbol{x}}_j^k) - \boldsymbol{\mu}_{i,j}^{\text{vec}}\right)^T \tag{6.17}$$

corrected for its biased expectation.



(a) Target image (b) Potential function (c) Combination

Figure 6.4: Qualitative illustration of the vector-evaluating potential function Eqs. (6.13) to (6.15) by assuming that both key points are the knees and fixing one position to the annotated position and varying the other one. (a) A sample target image with the annotated position for the right knee $\dot{x}_i$ marked with a "+". (b) Result of evaluating the potential function when varying the position of the left knee over the image domain while fixing the right one. (c) Modulation of the target image's transparency with the potential values to get a better intuitive understanding.

6.2 TERNARY SPATIAL STATISTICS

Rotation and scaling variance might be induced by the data, i. e., the anatomy, or by the acquisition and preparation of the data, e. g., varying images sizes or unconstraint camera positions. While the former is quite common, consider for example age-related anatomical changes, the latter one is mostly related to specific dynamic imaging modalities such as medical ultrasound. Since we are interested in the variability of the anatomy rather than the acquisition variability, we suggest two ternary potentials modeling the anatomical variability while being both rotation and scaling invariant.

6.2.1 *Distance Ratio*

The first ternary potential function evaluates the distance ratio of the two line segments created by three key points and uses a normal distribution to model it. The ratio is a quite reasonable feature in medical imaging, given that many parts of the human body change with a global parameter. Consider for example the lower and upper arm or the lower and upper leg, where both parts change over the course of immaturity while their ratio is mostly constant.

For three key points i, j and l, the potential function

$$\phi^{\text{dist}}_{i,j,l}(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l \mid \mu^{\text{dist}}_{i,j,l}, \sigma^{\text{dist}}_{i,j,l}) = f\left(d'(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l) \,\middle|\, \mu^{\text{dist}}_{i,j,l}, \sigma^{\text{dist}}_{i,j,l}\right) \tag{6.18}$$

computes the distance ratio

$$d'(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l) = \frac{d(\boldsymbol{x}_i, \boldsymbol{x}_j)}{d(\boldsymbol{x}_j, \boldsymbol{x}_l)} \tag{6.19}$$

and uses the probability density function of the normal distribution $f\colon \mathbb{R} \to \mathbb{R}$ (see Eq. (6.3)) to compute the probability density for the distance ratio. The normal distribution is parameterized by the mean distance ratio $\mu^{\text{dist}}_{i,j,l}$ and its standard deviation $\sigma^{\text{dist}}_{i,j,l}$.

The training is performed analogously to the training of the binary distance potential from Section 6.1.1, but using the distance ratio Eq. (6.19) as feature instead of the absolute distance Eq. (6.2) when estimating the sufficient statistics Eqs. (6.4) to (6.5).

6.2.2 *Relative Angle*

The second ternary potential function evaluates the relative angle between the two line segments created by three key points and uses a circular normal distribution to model it. For three key points i, j and l, the potential function

$$\phi^{\text{ang}}_{i,j,l}(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l \mid \mu^{\text{ang}}_{i,j,l}, \kappa^{\text{ang}}_{i,j,l}) = g\left(\alpha'(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l) \,\middle|\, \mu^{\text{ang}}_{i,j,l}, \kappa^{\text{ang}}_{i,j,l}\right) \tag{6.20}$$

first computes the relative angle

$$\alpha'(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_l) = \operatorname{sign}(\boldsymbol{v}_1 \times \boldsymbol{v}_2) \cdot \arccos\left(\frac{\boldsymbol{v}_1 \cdot \boldsymbol{v}_2}{|\boldsymbol{v}_1||\boldsymbol{v}_2|}\right) \tag{6.21}$$

between the projected (ρ; as in Section 6.1.2) difference vectors $\boldsymbol{v}_1 = \rho(\boldsymbol{x}_i - \boldsymbol{x}_j)$ and $\boldsymbol{v}_2 = \rho(\boldsymbol{x}_l - \boldsymbol{x}_j)$ of the key point positions $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ and $\boldsymbol{x}_l$ (see Fig. 6.5). Then, the probability density function of the circular normal distribution $g\colon \mathbb{R} \to \mathbb{R}$ (see Eq. (6.8)) is used to compute the probability density for the relative angle. The circular normal distribution is parameterized by the mean relative angle $\mu^{\text{ang}}_{i,j,l}$ and its concentration $\kappa^{\text{ang}}_{i,j,l}$. The necessity of explicitly defining a projection ρ for 3D images as in Section 6.1.2 still applies.

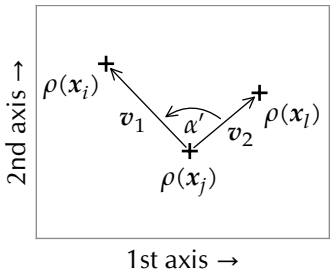


Figure 6.5: Illustration of the computed relative angle α' for the projected (ρ) difference vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ w. r. t. the three key point positions $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ and $\boldsymbol{x}_l$.

The training is performed analogously to the training of the binary angle potential from Section 6.1.2, but using the relative angle Eq. (6.21) as feature instead of the absolute angle Eq. (6.7).

6.3 SYMMETRY

All previously defined binary potential functions are symmetric w. r. t. the order of their arguments, i. e., the outcome of the potential functions for a key point pair is equal to the outcome of the respective potential functions for the reversed key point pair:

$$\forall \boldsymbol{x}_1, \forall \boldsymbol{x}_2 \colon \phi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi_{j,i}(\boldsymbol{x}_j, \boldsymbol{x}_i). \tag{6.22}$$

Hence, it is sufficient to use one binary potential function per key point pair instead of two instances, since they provide the same information and are identical in the eyes of the optimization process introduced in the next chapter.

In contrast, this does not hold true for the previously introduced ternary potential functions, where the order of the key points matters. Hence, the different orders introduce different kinds of potential functions. Using the optimization process that we'll introduce in the next chapter, the most relevant ones can be deduced automatically. However, it is also possible to use prior knowledge to only select reasonable arrangements.

6.4 LATENT GLOBAL TRANSFORMATIONS

A potential problem arises when an image at test time contains a constellation that is far off from the mode of estimated statistics. Consider for example outliers (e. g., extremely large or small patients) or simply wrong assumptions about the underlying distribution (e. g., the Gaussian assumption might not apply in every case), in which case the overall energy $E(\boldsymbol{s} \mid \boldsymbol{I})$ might be unproportionally high for the correct configuration.

To overcome this problem, a global affine transformation $\boldsymbol{T} \in \mathbb{R}^{D \times D}$ can be applied to all positions $\forall i \in [1 \,..\, N] \colon \boldsymbol{T}\boldsymbol{x}_{i,s_i}$ of the constellation $\boldsymbol{s}$ prior to using the positions for computing the relevant statistics. Note that this only applies to potential functions evaluating some sort of spatial statistic, i. e., it does not apply to the unary potential functions or potentially other types of higher order functions.

6.4.1 *Scaling*

The most interesting type of transformation is arguably the scaling transformation, as it is one natural global transformation of the human body caused by growth. The remaining transformations are of less interest due to the often fixed and rigid image acquisition scheme. Note that this is not always the cause though, consider for example fetal ultrasound imaging, where images pose more types of transformations

due to the unfixed position of the fetus in the womb. However, that kind of modality is not used here. Thus, we later evaluate the usage of isotropic scaling $T^{\mathrm{s}} \in \mathbb{R}$ as latent variable in terms of a global transformation $\forall i \in [1\,..\,N]\colon T^{\mathrm{s}}\boldsymbol{x}_{i,s_i}$ of the coordinate system.

6.4.2 *Abnormalities*

Another interesting application is the modelling of abnormalities. Although not done in this thesis, the general idea is outlined here as it might be of interest to the reader.

There are diseases that alter the standard constellation of key points. Consider for example scoliosis, which is a misalignment of the spine. While this can be captured to some extend in the spatial statistics, it is more of an outlier w. r. t. the standard alignment and might be treated as such. Assuming that the spatial statistics were estimated on healthy patients, the different degrees of scoliosis might be modeled in terms of a transformation back to the healthy alignment. This would not only allow to better handle cases affected by scoliosis, it would also automatically determine the degree of scoliosis.

7 OPTIMIZATION OF THE CRF

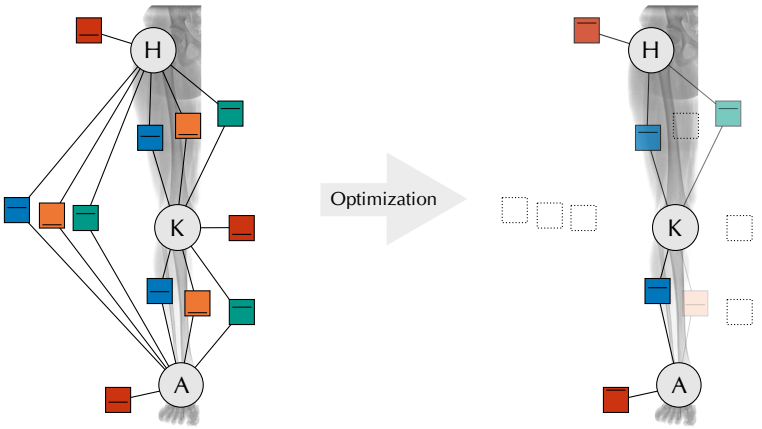
In the following, an approach is outlined to automatically estimate and optimize the graph topology, weighting the various potential functions and estimating the "missing" energies w. r. t. the detection and localization task on a specific dataset with a specific set of key points.

7.1 POOL OF POTENTIAL FUNCTIONS

A central problem of graph-based approaches that use non-unary potentials for spatial regularization is the definition of such potential functions. Additionally, it is not clear how these potential functions should be added to the factor graph in order to define a suitable graph topology w. r. t. the dataset and task at hand.

In Chapter 5, we illustrated how to derive unary potential functions from the local appearance models and in Chapter 6 we defined some simple binary and ternary potential functions utilizing probabilistic modelling of spatial features. However, there exist various other potential functions (e. g., those in [60, 16, 58, 59, 78, 157, 198, 4]) that were defined for different tasks and datasets that might be useful as well. Though, their definition and creation of the factor graph is often heuristically motivated and it is not clear which potential functions should be used and how they should be placed within the factor graph.

To this end, the concept of a *pool of potential functions* is proposed. Instead of heuristically selecting suitable potential functions and creating a "reasonable" graph topology, a fully loaded and fully connected graph should be created from the pool of potential functions Φ. Using this fully loaded graph, the task of the optimization is (1) to remove unnecessary potential functions and thereby defining a reasonable graph topology for the problem at hand, (2) to weight ($\boldsymbol{\Lambda}$) the individual potential function contributions and (3) to estimate the potential-specific "missing" energy values $\boldsymbol{B}$ w. r. t. the graph energy E. This idea is visually illustrated in Fig. 7.1 using a simplified hypothetical detection and localization task.



(a) Initial configuration (b) After optimization

Figure 7.1: Qualitative illustration of the CRF optimization process using the exemplary task of detecting and localizing the hip, knee and ankle. (a) The factor graph fully loaded with one unary potential function (■) per key point (depicted as circled H, K and A) and three different binary potential functions (■, ■ and ■) per key point pair prior to the optimization. Note that "missing" energies are depicted as horizontal lines in each factor node. (b) A hypothetical resulting factor graph after applying the optimization. Note that potential functions have been removed, weighted (illustrated by transparency) and the "missing" energies have been adjusted. For example, the relation between the hip and the ankle might not have provided useful information in contrast to the path over the knee and was thus removed, similarly to the unary potential of the knee. Note how such a simplification might create a tree-structured graph (after normalizing factors per clique), which greatly improves inference time in contrast to a fully connected graph. Note that this is just a hypothetical example to illustrate the basic idea.

7.2 LOSS FORMULATION

While there exist heuristic approaches (e. g., [16]) to estimate the potential weights and energies, the more common approach is to learn those parameters from data [107]. Here, the latter approach is followed by defining an appropriate loss function L computed over data $\mathcal{D}$ w. r. t. the detection and localization task and using gradient descent to minimize L.

The probabilistic approach is to use MLE, which requires the computation of the partition function Z (see Section 3.4.2). However, that becomes intractable quickly with increasing graph complexity (i. e., the topology in combination with the combinatorial complexity of the key points and the localization hypotheses) and stresses the influence of outliers. Furthermore, the configurations within the state space are highly correlated and only a subset of those represent important configurations to operate with.

Thus, a margin-maximizing approach in the energy domain is followed [117, 106, 107] that tries to increase the gap between the energy—recall Eq. (4.1)—of the "correct" configuration $\boldsymbol{s}_k^+$ and the best (i. e., lowest energy) "incorrect" configuration $\boldsymbol{s}_k^-$ for a training image k. The basic idea of that approach is that a change of the energy of the current *best* incorrect configuration applies to some extent to all other (having a higher energy) incorrect configurations due to the high correlation.

A well known loss function is the maximum-margin hinge loss [117], which tries to increase the energy gap between the configurations $\boldsymbol{s}_k^+$ and $\boldsymbol{s}_k^-$ until a certain margin m is satisfied. The intuition is that a margin m improves generalization and that only samples which do not satisfy the margin yet contribute to the loss. Let the loss function on K training samples $k \in [1\,..\,K]$ be defined as

$$\begin{aligned} L(\boldsymbol{\Lambda}, \boldsymbol{B}) = \frac{1}{K} \sum_{k=1}^{K} \Big[\delta(\boldsymbol{s}_k^+, \boldsymbol{s}_k^-) \cdot \max\big(0, m + E(\boldsymbol{s}_k^+ \mid \boldsymbol{\Lambda}, \boldsymbol{B}) \\ - E(\boldsymbol{s}_k^- \mid \boldsymbol{\Lambda}, \boldsymbol{B})\big) \Big] \\ + \theta \cdot \Omega(\boldsymbol{\Lambda}) \end{aligned} \tag{7.1}$$

subject to $\lambda_f \geq 0$ for all $f \in [1\,..\,F]$, with $\delta(\boldsymbol{s}_k^+, \boldsymbol{s}_k^-)$ weighting the K training configuration pairs and Ω regularizing the weights $\boldsymbol{\Lambda}$ with a regularization factor of θ.

The training samples are weighted by their reduction in error

$$\delta(\boldsymbol{s}_k^+, \boldsymbol{s}_k^-) = e(\boldsymbol{s}_k^-) - e(\boldsymbol{s}_k^+) \tag{7.2}$$

when going from the incorrect configuration $\boldsymbol{s}_k^-$ to the correct configuration $\boldsymbol{s}_k^+$. The corresponding error of a configuration

$$e(\boldsymbol{s}) = \frac{1}{NR} \sum_{i=1}^{N} \begin{cases} 0, & \text{if } s_i = 0 \text{ and the } i\text{-th key point is missing (or no correct hypothesis exists in } \mathcal{X}_i) \\ \min\left(\|\overset{\star}{\boldsymbol{x}}_i - \boldsymbol{x}_{i,s_i}\|_2, R\right), & \text{if } s_i > 0 \text{ and the } i\text{-th key point exists} \\ R, & \text{otherwise,} \end{cases} \tag{7.3}$$

is defined as the normalized (i. e., $e \in [0, 1]$) summation of all key-point-specific errors, which are either given by a detection error penalty R or the capped Euclidean distance between the annotated key point position $\overset{\star}{\boldsymbol{x}}_i$ and a corresponding predicted one $\boldsymbol{x}_{i,s_i}$ (in case of a true positive detection). Weighting the samples helps to steer the joint optimization of $\boldsymbol{\Lambda}$ and $\boldsymbol{B}$ by first focusing on coarse (detection) and then fine (localization) details.

In case refinement as outlined in Section 4.4 is going to be used, the first branch of the error function Eq. (7.3) uses an additional condition (written in parenthesis in Eq. (7.3)) that ensures that selecting the "refine" label is not penalized in case no correct localization hypothesis exists.

In addition to the K data-dependent weighted loss terms, a θ-weighted regularization term of the potential weights $\Omega(\boldsymbol{\Lambda})$ is added. While an L^1 penalty $\Omega(\boldsymbol{\Lambda}) = \|\boldsymbol{\Lambda}\|_1$ is commonly seen in the literature [106] to accelerate the sparsification of terms (i. e., $\lambda_f = 0$), an L^2 penalty $\Omega(\boldsymbol{\Lambda}) = \|\boldsymbol{\Lambda}\|_2$ or no regularization $\Omega(\boldsymbol{\Lambda}) = 0$ are also valid options.

Note that although no potential function parameters are optimized in this loss function, it is theoretically possible to do so for any kind of differentiable potential function w. r. t. their parameters (cf. [186]). An adapted formulation to do so as well as an initial experiment is presented in Appendix A.

7.3 RIVAL SELECTION

The definition of a correct configuration $\boldsymbol{s}_k^+$ for a training sample is straightforward w. r. t. the detection task. For the localization task however, a more elaborate definition is useful. Instead of using the annotated position $\overset{\star}{\boldsymbol{x}}_i$ for each key point i, the closest corresponding localization hypothesis $\arg\min_j \|\overset{\star}{\boldsymbol{x}}_i - \boldsymbol{x}_{i,j}\|$ is used to render the optimization more approachable, i. e., $\boldsymbol{s}_k^+ = \arg\min_{\boldsymbol{s}} e(\boldsymbol{s})$. If there is no localization hypothesis within R for any key point, the current training sample is removed from the training set (this is rarely the case though).

Note that this is not done when refinement as outlined in Section 4.4 is going to be used, in which case the "refine" label is treated as the correct value for the respective key points.

For the rivaling configuration $\boldsymbol{s}_k^-$, CRF inference is used to find the best incorrect configuration

$$\begin{aligned} \boldsymbol{s}_k^- &= \arg\min_{\boldsymbol{s}} E(\boldsymbol{s} \mid \boldsymbol{\Lambda}, \boldsymbol{B}) \\ &\text{subject to } e(\boldsymbol{s}) > e(\boldsymbol{s}_k^+). \end{aligned} \tag{7.4}$$

Since the CRF inference in Eq. (7.4) is not differentiable w. r. t. the optimized parameters, the rival $\boldsymbol{s}_k^-$ is treated as constant in the loss formulation Eq. (7.1) while running the inference again with each parameter change. With this specification, the loss formulation in Eq. (7.4) becomes a non-convex optimization problem. To solve Eq. (7.4) exactly and generally, A* is applied again (to find the m-best configurations [11]). Note that various other algorithms are applicable as well, potentially providing better runtime performance (see Section 3.4.3). Alternatively, Markov chain Monte Carlo (MCMC) methods can be used when inference becomes intractable to find a suitable rival $\boldsymbol{s}_k^-$ [93]. Note that the CRF inference is practically more feasible than the estimation of the partition function.

7.4 OPTIMIZATION VIA STOCHASTIC GRADIENT DESCENT

The optimization of the loss function L from Eq. (7.1) is carried out using stochastic gradient descent [21, p. 240] in form of the Adam algorithm [104]. It has shown great success in various optimization problems while requiring little parameter tuning. I. e., using the default parameters proposed in [104] and simply altering the global step size (learning rate) η (α in [104]) already provides reasonable performance in various tested machine learning settings.

The optimization is started with the initial weights set to $\boldsymbol{\Lambda}^0 = 1$, which corresponds to a formulation without weights and a fully connected and fully loaded graph. For the energies $\boldsymbol{B}$, a more elaborate scheme to estimate the initial values is used in order to provide values in the correct magnitude w. r. t. the potential function values. Thus, energies are computed for all potential functions in Φ—separately for all types of potential functions—over all correct configurations $\boldsymbol{s}_k^+$ if no key point in the scope of the potential function is labeled "missing". From the sets of potential-specific estimated energies for correct clique configurations the 85-th percentiles are estimated and used as respective initial energies $\boldsymbol{B}^0$ (see Fig. 4.2 for an intuitive explanation).

Finally, the potential weights $\boldsymbol{\Lambda}$ and the "missing" energies $\boldsymbol{B}$ are iteratively refined using a mini-batch of size $K = 40$ training images per

iteration w. r. t. the detection and localization task until convergence. By removing zero-weighted potential functions (i. e., $\lambda_f < 1\text{E}{-}10$), the factor graph is accelerated and the graph topology is geared towards the dataset at hand, removing the need for heuristically selecting potential functions.

7.5 IMPORTANT PARAMETERS

While the used values for most parameters already provide reasonable default values, the importance of some parameters is stressed again in the following listing:

SAMPLE WEIGHT Given that the objective being optimized is non-convex, the trajectory of the optimization is of great importance. Even more so when optimizing the weights $\boldsymbol{\Lambda}$ jointly with the "missing" energies $\boldsymbol{B}$, which might be very off initially. The result might be the selection of incorrect rivaling configurations that are of less importance to the detection and localization target. Thus, a weighting of the samples is of utter importance to achieve reasonable performance by reaching a good local minimum. The provided weighting δ has shown to provide generally good results. Note that the sample weighting can also be used to place emphasis on different tasks (e. g., detection more important than localization or vice versa) or different key points (e. g., important versus auxiliary) by defining appropriate error functions.

ENERGY MARGIN A proper selection of the energy margin m, which effectively filters the training samples, can improve the generalization capabilities. The value however depends on the graph parameterization and should be selected in a cross-validation setup.

WEIGHT REGULARIZATION A regularization of the weights using an L^1 or L^2 penalty (with an influence of θ) might improve the results even further. Although we did not obtain conclusive results which penalty should be preferred, the results indicated that regularization is generally useful and that the type of regularization should be evaluated in a cross-validation setup.

TRAINING DATA For better convergence, the optimization should be carried out on a different subset of training data than was used to train the potential functions Φ (local appearance models plus spatial regularization). This can be a drawback if the set of training images is very small. However, we later illustrate that—depending on the target objects—it is possible to successfully train a model in such a case as well.

8

3D CONVOLUTIONAL POSE MACHINE

In contrast to explicitly modelling the correlation between key points, a recent trend is the usage of CNNs to automatically engineer the features necessary to exploit the spatial correlation [203, 150]. Given the recent success of such methods in the medical domain [20, 151], a comparison to such a method is very interesting. Thus, the convolutional pose machine (CPM) architecture—which showed great performance on non-medical [203] as well as medical data [20]—is used as method to compare against for a more detailed evaluation w. r. t. the exploitation of the key point correlation, in addition to the comparison of published results of other methods.

8.1 PROBLEM SETTING

As the construction of a well-performing CNN architecture is also dataset dependent (e. g., necessary receptive field, dataset size in contrast to hardware limits), we focus on a concrete, relevant problem which has not been tackled so far: the localization of many—i. e., more than 100—key points in CT scans of the spine using a CNN.

Settling for a concrete problem allows to tune the respective methods towards the dataset for better comparability, which is especially useful when it comes to CNN architectures that regress heatmaps for very large amounts (≥ 100) of key points. Often, the commonly used encoder-decoder architectures (e. g., U-Net [164, 45] and V-Net [139]) that produce heatmaps as large as the input image are not feasible to be trained using a recent GPU with, e. g., 12 GB of memory. For example, assuming a somewhat representative image size of $512 \times 512 \times 200$ vx for a spine CT scan and 100 target key points, solely the input image plus the target heatmaps would require minimally $\sim$ 21 GB of GPU memory (not including any gradients or other required outputs). Even for downsampling architectures as the CPM, the training for such large images is not possible, which is why the training is often performed on patches cut from the original images that are a fraction of the original size. However, this is highly dataset and target object dependent. Furthermore, long-range dependencies can not be learned as they are never seen in training. Thus, a comparison between the previously introduced method and such a method is of great interest.

While the respective dataset is introduced in detail later in Section 10.5, a few parameters are mentioned here to motivate the following. The goal is to localize 102 key points, 6 for each of the 17 vertebrae T1 to

L5 (see Fig. 10.11). The sagittally sliced images were downsampled to an anisotropic resolution of $3\times1\times1\,\mathrm{mm}^3/\mathrm{vx}$ (illustrated as left-right × posteroanterior ×superoinferior).

8.2 CNN ARCHITECTURE

As hinted earlier, the correlation-exploiting convolutional pose machine (CPM) architecture is used as basis for the dataset-specific optimization. It has shown to provide very good results in medical [20] and non-medical [203] settings and is—in contrast to an encoder-decoder CNN architecture—actually trainable for 102 key points with a recent GPU due to the reduced memory demand (still requiring a tiling of the input image, though). As proposed by Wei et al. [203], the CPM is a fully convolutional neural network architecture to regress downsampled key-point-specific heatmaps. Key concept of this architecture is the stacking of similar prediction modules, where intermediate modules get the image and the heatmaps of the previous module in order to solve confusions between key point predictions by looking at the predictions of adjacent key points. The basic idea is that the network implicitly learns and exploits the co-occurrence of different key points. In combination with the later suggested interpolation approach to counter the reduced accuracy caused by the missing upsampling (which is not possible due to the exhaustive memory demand), this approach is arguably the best choice for the dataset at hand with that many key points.

8.2.1 *Modifications*

Here, the originally 2D approach [203, 20] is extended to 3D image data and the network architecture is adapted to match the anisotropic resolution ($3\times1\times1\,\mathrm{mm}^3/\mathrm{vx}$) of the spine dataset. The resulting architecture—referred to as "3D CPM"—is illustrated in Fig. 8.1. An input patch volume of $32\times96\times256$ vx (roughly $10\times10\times25\,\mathrm{cm}^3$) is used, which is (a) large enough to exploit the co-occurrence of neighboring—especially along the superoinferior axis—vertebra key points (see input volume in Fig. 8.1) and (b) small enough for the model (producing heatmap volumes for 102 key points) to be fully trainable with at least 3 stages on a GPU with 12 GB of memory. The pooling configurations and kernel sizes are chosen such that the receptive field size slightly exceeds the input patch size while keeping the anisotropic voxel size in mind. The proposed network architecture can be used for different resolutions as well, either by resampling the input or by modifying the input patch size and the filter sizes. The latter is arguably less practical though and makes such dataset-specific networks not easy to transfer to different datasets in contrast to the previously introduced CRF-based method.

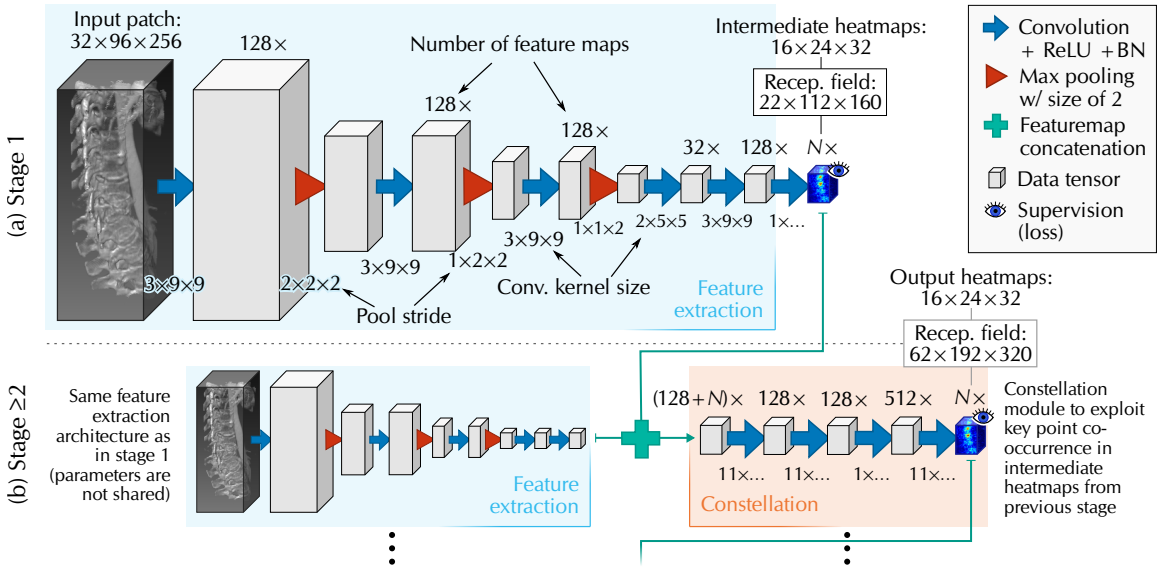


Figure 8.1: Illustration of the 3D CPM architecture as used to localize many key points in anisotropic CT scans of the spine. The feature extraction module used in the (a) first stage is used again in (b) successive stages in combination with a constellation module. Note that parameters are not shared and more than two stages can be used.

At test time, input patches from the test image are extracted using an overlapping sliding window approach—the overlap being averaged—with a stride of $1/3$ of the input patch size. The overlap is necessary to counter the exponential reduction in the receptive field strength [129] towards the patch border as well as the usage of zero-padded convolutions.

8.2.2 *Training*

A proper selection of training patches is crucial for good performance of the trained network. From each training image, we randomly cut 12 input patches of size 32×96×256 vx with the constraint to contain at least one key point. Empirically, each resulting input patch contains more than 10 key points, which provides enough co-occurrence clues to be exploited. For each input patch, we create 102 key-point-specific target heatmaps placing a Gaussian kernel at the true position of the key points contained in the patch (similarly to the previous methods). We use the summation of multiple SSE functions—one for each stage's output mimicking intermediate supervision—as loss function and minimize it using SGD in form of the Adam optimizer [104].

8.3 POLYNOMIAL REFINEMENT

The downsampling architecture of the 3D CPM allows to target localization tasks with many key points, but at the price of highly quantized

output heatmaps and thus comparably imprecise localizations. For instance, the resolution of the such generated heatmaps is $6{\times}4{\times}8\,\mathrm{mm}^3/\mathrm{vx}$, which is caused by the downsampling by a factor of $2{\times}4{\times}8$ (in each dimension). Which means that even "hitting" the correct voxel does not translate to a precise physical position and that being off—even by just one voxel—might cause a mis-localization.

To efficiently compensate for such errors, a fast interpolation technique based on a 3D second order polynomial (cf. [194]) is suggested. It is the simplest interpolation matching the expected bell-like shape of the heatmap maxima caused by the Gaussian target kernel. More concretely, we assume that the polynomial

$$u(x,y,z) = a_1x^2 + a_2y^2 + a_3z^2 + a_4xy + a_5xz + a_6yz + a_7x + a_8y + a_9z + a_{10} \quad (8.1)$$

can locally interpolate the values around the maximum $\boldsymbol{x} = (x\ y\ z)$ within a 1-voxel neighborhood w. r. t. the coefficients $\boldsymbol{a} = (a_1 \dots a_{10})$. The coefficients $\boldsymbol{a}$ are estimated by using ordinary least squares (closed-form solution) with the maximal heatmap value plus the heatmap values in an 26-connected voxel neighborhood (8-connected pixel neighborhood in 2D) around the maximum pixel position as bases. Then, the continuous position of the maximum of the estimated polynomial is analytically computed and the necessary conditions for the stationary point to form a maximum are verified. Additionally, the maximum is discarded if it is outside of the local neighborhood. Otherwise, this corresponds to the *refined* position of the discrete local maximum $\boldsymbol{x}$ and is used instead of it. This process is applied independently for each heatmap to the respective quantized global maximum.

This approach translates analogously to 2D heatmaps using a 2D second order polynomial and following the same protocol. A visualization of this refinement in 2D can be seen in Fig. 8.2.

Furthermore, this refinement may also be useful if no downsampling happens, but if the resolution is highly anisotropic or very small in general. Thus, it can be used with any kind of heatmap-generating method that is geared towards generating bell-like local maxima.

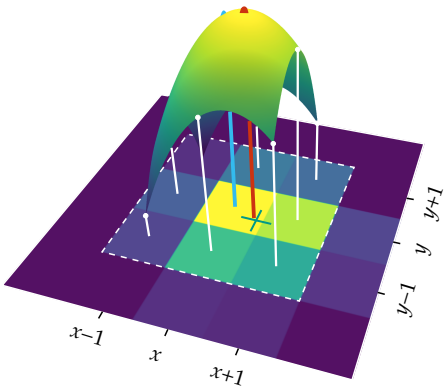


Figure 8.2: Illustration of the polynomial refinement applied to a artificial 2D heatmap (bottom) generated by placing a Gaussian kernel at $(x + 0.4,\ y - 0.2)$ (marked by a ■ cross). The estimated polynomial is visualized in combination with the bases (white lines), the heatmap maximum position (■ line) and the analytical maximum of the polynomial (■ line). Note how close the refined position is to the Gaussian mean.

Part III

EXPERIMENTAL SETUP

9 IMPLEMENTATION

In the following, the software stack used to implement the previously introduced method is described. Additionally, the hardware configuration on which the software was run for the subsequent experiments is reported.

9.1 SOFTWARE

The method has been implemented in Python 3.6, since it is freely available[1] and provides a rich scientific ecosystem of readily available software packages.[2] Its dynamic nature allows for rapid prototyping at the cost of efficient runtime performance compared to, e. g., compiled languages like C or C++. Thus, this implementation should be considered non-optimized research code.

Table 9.1: List of major third-party Python packages that have been used. This listing includes the most recent versions used as well as a short description of the major functionality provided and used.

PACKAGE	VERSION	PROVIDES
ITK	5.0.1	Medical image representation, serialization and common image operations [138]
numpy	1.16.4	Efficient n-dimensional data array representation and common mathematical routines [199]
OpenGM	unreleased*	Efficient factor graph representation and various inference algorithms [5]
scikit-learn	0.21.3	Various machine learning algorithms such as decision trees [152]
scipy	1.3.3	Common statistical functions as well as common image operations [196]
SimpleITK	1.2.4	Simplifying wrapper around ITK (mostly used instead of ITK if possible) [127]
tensorflow	1.13.1	Compute-graph representation with automatic differentiation used for CNNs [1]

* An unreleased version with custom Python 3 support has been used. The code is publicly available at https://www.aomader.com/phd/.

[1] Software packages for different versions of Python are publicly available at https://www.python.org.

[2] The main repository for Python packages is publicly located at https://pypi.org/ and provides a rich set of scientific software packages.

While there are a lot of (transitive) third-party dependencies, only a few—listed in Table 9.1—provide the actually necessary algorithms that have been used in this implementation. Since the evaluation was performed over a prolonged period of time, different versions of those packages have been used. Thus, this list contains only the most recent versions used. Note, this also applies to the usage of different minor versions of Python 3.6.

A snapshot of the used implementation is publicly available at `https://www.aomader.com/phd/` for other researchers to inspect, use and improve.

9.2 HARDWARE

The experiments performed during the evaluation were conducted on three different types of headless servers, whose specifications are listed in Table 9.2.

Table 9.2: List of machines and their specifications that were used to conduct experiments. All machines used a RAID1 setup of conventional hard drives for data storage.

MACHINE	HARDWARE SPECIFICATION		
	CPU	GPU	MEMORY
alpha	2 × Intel Xeon E5-2650 v4	2 × NVIDIA TITAN X	512 GB
beta	2 × Intel Xeon E5-2650 v4	2 × NVIDIA TITAN Xp	512 GB
gamma	2 × Intel Xeon Silver 4210	4 × NVIDIA TITAN RTX	748 GB

Note that these are shared servers, thus the resources have not been used exclusively and the actually available amount of memory and CPU power at the time of the experiments might have been less than stated in Table 9.2.

10 DATASETS

The proposed method has been evaluated on various medical datasets targeting different anatomical structures. In addition to showing different regions of the human body, the datasets also differ in the number of key points (N), the image dimensionality (2D and 3D), the image resolution (mm/vx; isotropic and anisotropic) and imaging modality (X-ray, CT and MRI) in order to illustrate the transferability of the proposed method. In the following, a short characterization is given for each dataset including a description of the imaged anatomy, the target key points as well as the challenges that arise for the task. Note that image sizes are given in pixel (px) if it is a 2D dataset (i. e., $D = 2$) and in voxel (vx) if it is a 3D dataset ($D = 3$).

10.1 LEFT HAND RADIOGRAPHS

The first dataset consists of 805 clinical radiographs of the left hand from the University Hospital RWTH Aachen and the University Medical Center Schleswig-Holstein initially provided by Dr. Kayser [86, 85, 28]. The 805 patients, 41.6 % of whom are female, were in the age of 3 to 19 years at the time of the image acquisition (see Fig. 10.1).

Anatomy

All except one image show the left hand (one showing the right hand) as well as the connection to the forearm. The images were acquired with a posteroanterior (PA) projection and, according to radiological protocol, are centered around the third metacarpal head while extending lateral to the skin margins, proximal to include the distal radioulnar joint and distal to the tips of the distal phalanges.

The images do not come with a physical resolution, are however isotropic and have an average size of 1212×2148 px (ranging from a minimum of 804×1347 px to a maximum of 1955×2964 px).

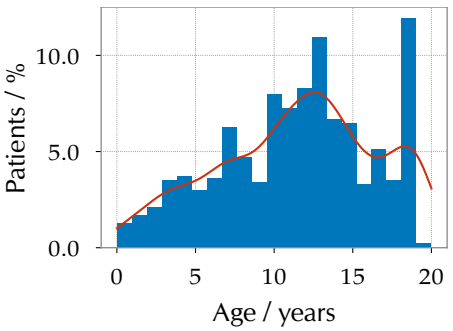


Figure 10.1: Distribution of the patient's age in the hand radiographs dataset.

Task

Hand radiographs are often used for the assessment of the bone age of immature patients. A key indicator in such radiographs is the gap of the epiphyseal plates of the finger joints. These plates start with a gap that slowly narrows till maturity, which can be used for an automatic assessment of the bone age [156]. Hence, the task is to localize the three epiphyseal plates of each of the four fingers contained in the radiographs. Given that all epiphyseal plates are always visible, this dataset does not pose the detection problem but just the localization problem. For the latter, Fischer et al. [68] introduced a relative localization threshold—i. e., the maximal distance between the annotated and predicted key point position to treat a key point as correctly localized—of $6/256$ of the image's height (see last image in Fig. 10.2). In order to maintain comparability, the same threshold is used here.

Challenges

The core challenge of this dataset is the repetitive nature of the finger joints, which often causes confusion between joints due to the local ambiguity. Furthermore, the dataset presents strong age-related variability due to the growth rate of the immature patients (3 to 19 years). Although the acquisition regime is fixed, we have pose variability w. r. t. the spread of the fingers, which can be classified into fully spread and not spread, as well as the orientation of the hand due to rotated images. A small fraction of radiographs also shows implants and some other images suffer from artificially added text on top of the radiographs after acquisition.

Preparation

Hahmann et al. [86] split the dataset into two subsets of 400 training and 412 test images. In addition to the one image showing a right hand, it has been noticed that six images were wrongly duplicated. These images have been excluded in order to create a homogenous and non-repetitive dataset, resulting in 395 training and 410 test images. To speed up the processing, the images have been resampled to a constant height of 600 px while maintaining the aspect ratio.

From now on, this dataset is referred to as "hands". A few images from this dataset including the corresponding annotations and localization threshold are depicted in Fig. 10.2.

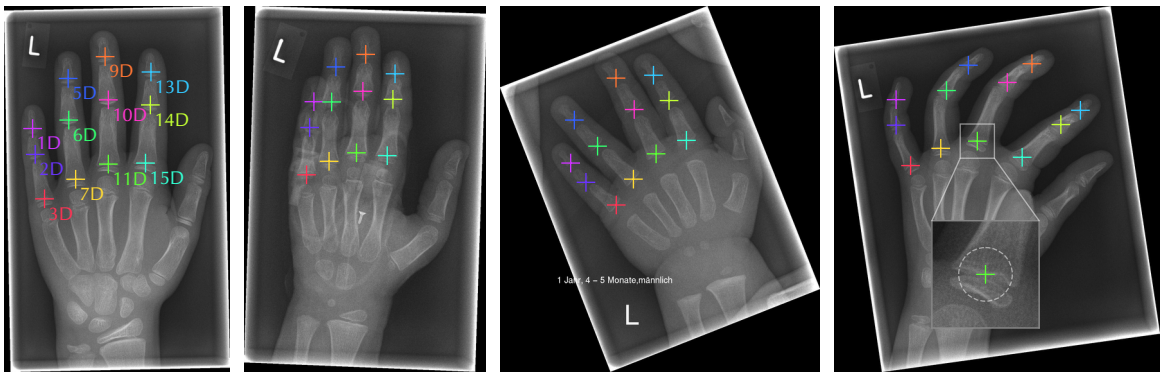
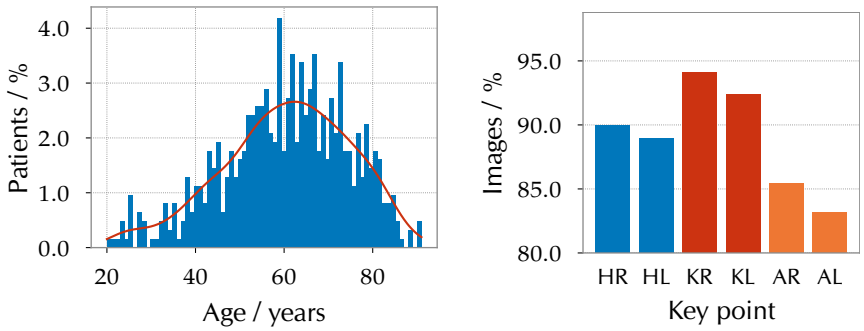


Figure 10.2: Illustration of four images of the hands dataset. The images are overlayed by the corresponding labeled (1D–3D, 5D–7D, 9D–11D, 13D–15D) key point annotations. Note the different spread between the fingers and the age-related structure of the finger joints (the age of the patients at acquisition time was, from left to right, 9, 3, 1.5 and 8 years). The localization threshold is illustrated as circle depicted in the inset in the last image.

10.2 LOWER LIMB RADIOGRAPHS

The second dataset consists of 660 digital radiographs of the lower extremities acquired from 606 patients. The 606 patients, 58.3 % of whom are female, were in the age of 20 to 91 years a the time of the image acquisition (see Fig. 10.3a). Of 52 patients, multiple images were acquired for pre- and post-operative assessment after inserting prostheses.

The images were acquired by a digital X-ray system with an exposure size of 3000×3000 px (illustrated as left-right ×superoinferior) at an isotropic resolution of 0.143×0.143 mm^2/px, providing 429×429 mm^2 spatial extent with each exposure. Since this area is not large enough to capture the complete lower extremities, multiple exposures at different positions are stitched together to form one joint radiograph [81].



(a) Age distribution (b) Amount of key points

Figure 10.3: Illustration of dataset parameters of the lower extremities: (a) Distribution of the patient's age with a mean age of 60 years. (b) Amount of hip (H, ■), knee (K, ■) and ankle (A, ■) key points of both sides (R = right, L = left) contained in all images. Note that more right legs are contained and the knee is contained most often.

Anatomy

All images show the lower extremities acquired by an anteroposterior (AP) projection. The field of view varies from full extent, i. e., fully covering both legs plus the hip region, to only parts of that area, e. g., only the right or left leg, one or both thighs, one or both lower legs, etc. In addition to the normal anatomy, the images also show various implants like knee replacements, femur joint replacements and fixations consisting of nails and pins.

The images also feature an artificial ruler placed in-between the patient and the X-ray detector along the superoinferior direction and (most of the time, but not always) midsagittal, which is visible in all images. The ruler is used by the stitching algorithm as a feature for reconstruction.

The images have an average size of 3016×7456 px (ranging from a minimum of 2600×4111 px to a maximum of 3074×8585 px), while covering an average physical extent of 431.3×1066.3 mm^2 (ranging from 371.8×587.9 mm^2 to 439.6×1227.7 mm^2).

Task

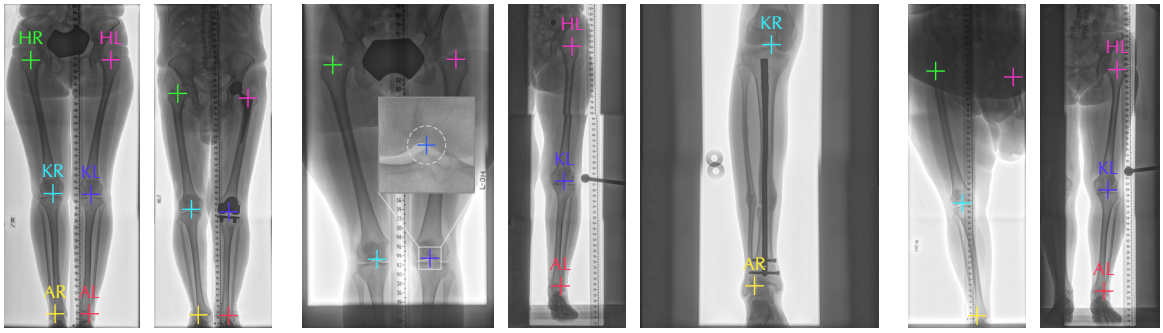
The task is to detect and localize (if present) the hip, knee and ankle joint of both legs in order to initialize a model-based segmentation [80] for orthopedic measurements.

Each key point was annotated twice by one observer, resulting in an intra-observer error of 2.3 mm, 1.3 mm and 2.6 mm for the hip, knee and ankle, respectively [168]. In order to consider a key point prediction correct, Ruppertshofen et al. [168] established a maximal Euclidean distance between the annotated and the predicted key point position of 10 mm (see third image in Fig. 10.4).

Due to a restricted field of view and missing limbs in a subset of images, only 73.4 % of the images contain all key points. Of the remaining images, 8, 78, 77 and 10 images only contain 5, 4, 3 and 2 key points, respectively. The distribution of annotated key points over all images is depicted in Fig. 10.3b. Furthermore, 11.3 % of the target key points are altered by prostheses or pathologies.

Challenges

The detection problem is mostly influenced by the highly varying field of view showing different parts of the anatomy. This is further complicated by missing limbs that would normally be contained in the respective field of view.



(a) Full lower body (b) Restricted field of view (c) Missing

Figure 10.4: Illustration of seven images of the legs dataset. The images are overlayed by the corresponding labeled key point annotations (color-coded "+"), i. e., right and left hip (HR, HL), right and left knee (KR, KL) and right and left ankle (AR, AL). (a) Examples of images containing all 6 key points. Note the knee joint and femur ball joint replacements in the second image. (b) Examples of images showing only subsets of the 6 key points. Note the image artifact in the second image and the implant in the third image. The localization threshold of 10 mm is illustrated as circle in the inset in the first image. (c) Examples of missing key point annotations due to a missing limb (first image) and low image contrast (second image).

Additionally, the anatomy itself features a lot of pathologies, including implants in form of prostheses of the knee and femur as well as pins and nails. But also fractures (fixated and not fixated ones) are contained, in addition to abnormal alignments of the legs.

Furthermore, due to the stitched reconstruction of a joint radiograph, image parts differ in contrast, with sometimes lower contrast in the upper body. The stitching algorithm may also introduce artifacts in form of streaks in case of strongly varying values. The required ruler may also cover the key point area.

It has also been observed, that the ankle is often very close to the image boundary, which often shows a gradient with very few contrast.

Preparation

The original dataset split used in [168] could not be reproduced, thus a patient-grouped (i. e., ensuring that all images of a patient are either in test or training but not in both) 5-fold cross-validation setup has been used. This resulted in, on average, 538 training images per fold. Note that this better excludes the effect of the chosen dataset split w. r. t. the originally used setup. To speed up the processing, the images have been resampled to an isotropic resolution of $1 \times 1\,\mathrm{mm}^2/\mathrm{px}$.

From now on, this dataset is referred to as "legs". A few images from this dataset including the corresponding annotations and localization threshold are depicted in Fig. 10.4.

10.3 THORAX RADIOGRAPHS

The third dataset consists of 642 radiographs of the thorax taken from the publicly available anonymized Indiana chest X-ray collection from the U. S. National Library of Medicine [55, 56]. There is exactly one radiograph per patient.

Anatomy

The Indiana chest X-ray collection contains PA-projected X-ray examinations. These examinations come with a report and a radiograph showing the lungs, extending left-right and superior towards the shoulders and inferior towards the abdomen. The images have been further altered by overlaying—thus removing—identifying objects like jewelry. Furthermore, they also show implanted medical devices, surgical devices and disease-inflicted abnormalities.

The images have an average isotropic resolution of $0.147 \times 0.147\,\mathrm{mm}^2/\mathrm{px}$ (illustrated as left-right×superoinferior; ranging from a minimum of $0.1 \times 0.1\,\mathrm{mm}^2/\mathrm{px}$ to a maximum of $0.2 \times 0.2\,\mathrm{mm}^2/\mathrm{px}$) and an average size of $2801 \times 2648\,\mathrm{px}$ (ranging from $1760 \times 2016\,\mathrm{px}$ to $4248 \times 4248\,\mathrm{px}$), while covering an average physical extent of $410.1 \times 387.5\,\mathrm{mm}^2$ (ranging from $285.1 \times 302.3\,\mathrm{mm}^2$ to $430.1 \times 432.0\,\mathrm{mm}^2$).

Task

In standard radiological practice, rib counting is used to assure a proper inhalation state in chest X-ray quality assessment. Thus, the task is to localize and label the posterior ribs in chest radiographs. Before our work in [130], a robust anatomical correct labeling of posterior ribs in chest radiographs has not been demonstrated. There is a number of methods described in literature to segment ribs in chest radiographs using either pixel classification [125, 204], atlas registration [35], or a mixture of methods [15]. Even the atlas-based method did not use rib labels. Recently, the extended abstract by Wessel et al. [204] illustrated a labeling based on bounding box detections.

Since the radiographs do not come with rib labels nor key point annotations, a semi-automatic approach was used to generate those for the unlabeled images. First, 1000 consecutive images from the Indiana chest X-ray collection were taken. For those 1000 images, unlabeled centerline annotations of the posterior ribs were automatically generated using a method based on [15] (dynamic programming in a summed response map of gradient-based and Hessian-based filters with ad-hoc extensions). After that, the centerlines were manually labeled and checked for quality, i. e., centerlines for the posterior ribs 2 to 9 (left and right) should be present and the lines should properly follow the

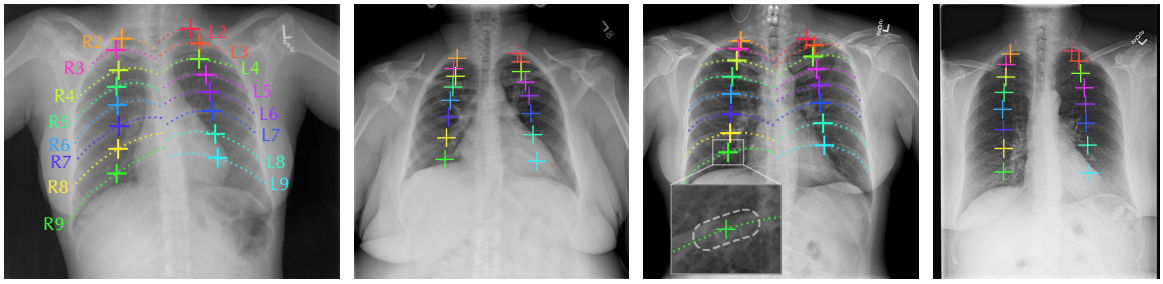


Figure 10.5: Illustration of four images of the chests dataset. The images are overlayed by the corresponding labeled (R1–R9 and L1–L9; color-coded) key point annotations as well as the corresponding rib centerlines. The centerlines are not shown in the second and fourth image in order to not obfuscate the low contrast of the ribs. The localization criterion is illustrated as outline in the inset in the third image.

center of the respective rib. The midsagittal points on the centerlines have been selected as point annotations for each key point (except for the second and third rib where a factor of ± 0.3 and ± 0.4, respectively, instead of 0.5 was chosen to compensate for the different anatomical structure of those ribs). This resulted in 642 radiographs annotated with 16 labeled posterior rib centerlines and corresponding key point annotations.

The such generated key point annotations allow to cast the rib labelling problem to a key point localization problem. In order to consider a predicted key point position correct (localization and labeling criterion), it has to be close to the annotated position (Euclidean distance $\leq 15\,\text{mm}$) and very close to the respective annotated centerline (minimal Euclidean distance $\leq 7.5\,\text{mm}$). This resembles the test whether the point lays on the rib while allowing for some translation along the rib. This localization criterion is illustrated in the third image in Fig. 10.5.

Challenges

Similar to the hands dataset, the thorax poses the problem of repetitive target objects and possible confusion, given that posterior ribs (as well as the anterior ribs) look locally very similar (except for the first two ones). Additionally, the artificially generated key point annotations are not well-defined anatomical landmarks and pose an inherent inaccuracy despite having been manually checked for quality. Hence the specifically crafted localization criterion to allow for some translation along the rib centerline.

Another challenge is generally the image quality and the associated image contrast. This is further reinforced by the artificial removal of jewelry and potentially important structures, as well as medical implants and generally different body poses which may generate very different shadows due to the different beam attenuation. The clavicle is

a notable structure to cause confusion due to shadowing the ribs, which is not consistent due to variation in pose.

Furthermore, low contrast in the region of the abdomen in combination with very few meaningful surrounding structures renders this part of the thorax very hard to analyze even for humans. Unlike in CT where this task could be solved easier [205], the upper ribs are often overlaid in a chest radiograph (by, e. g., clavicles and other ribs) in a way that may prevent an algorithm from identifying and counting all the upper ribs properly. Also using the lung field as reference space appears not to be sufficient to unambiguously assign an anatomic label to a detected rib.

Preparation

In order to follow best practices, the 642 radiographs were split into 3 folds in order to perform cross-validation. This resulted in 428 training images per fold. To speed up the processing, the images have been resampled to an isotropic resolution of $1 \times 1\,\mathrm{mm}^2/\mathrm{px}$.

From now on, this dataset is referred to as "chests". A few images from this dataset including the corresponding annotations and the localization criterion are depicted in Fig. 10.5.

10.4 SPINE SECTION CT SCANS

The fourth dataset consists of 302 public[1] CT scans of different spine sections as used in the MICCAI CSI 2014 challenge [78] that have been acquired at the Department of Radiology at the University of Washington. Note that various patients were scanned multiple times and thus multiple scans per patient exist.

Anatomy

All images show some part of the spine (i. e., a section) including variable amounts of vertebrae. The images contain highly pathological cases showing, e. g., high grade scoliosis, kyphosis, fractures and numerous surgical implants. Additionally, the images were altered by masking the spine using a cylinder and setting values outside the masked area to a constant value of −3020 HU. An axial slice of a random image is depicted in Fig. 10.6 illustrating the circular mask.

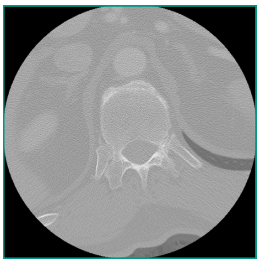


Figure 10.6: Illustration of the cylinder mask that has been applied to the CT scans showing an axial slice.

[1] The images are accessible via the SpineWeb collaboration platform located at http://spineweb.digitalimaginggroup.ca.

The images are axially sliced and have an average anisotropic resolution of $0.336 \times 0.336 \times 2.059\,\mathrm{mm}^3/\mathrm{vx}$ (illustrated as left-right ×posteroanterior ×superoinferior; ranging from a minimum of $0.195 \times 0.195 \times 0.625\,\mathrm{mm}^3/\mathrm{vx}$ to a maximum of $0.508 \times 0.508 \times 2.5\,\mathrm{mm}^3/\mathrm{vx}$) and an average size of $512 \times 512 \times 160$ vx (ranging from $512 \times 512 \times 31$ vx to $512 \times 512 \times 511$ vx), while covering an average physical extent of $172.4 \times 172.4 \times 318.0\,\mathrm{mm}^3$ (ranging from $100.0 \times 100.0 \times 63.1\,\mathrm{mm}^3$ to $260 \times 260 \times 710\,\mathrm{mm}^3$).

Task

Various diseases and pathologies with a high impact on life quality affect the spine. This ranges from deformations like scoliosis, over pain inflicting tissue degradation to fractures and osteoporotic changes. Even more so in trauma settings, where an accurate and fast evaluation of the spine is of major importance.

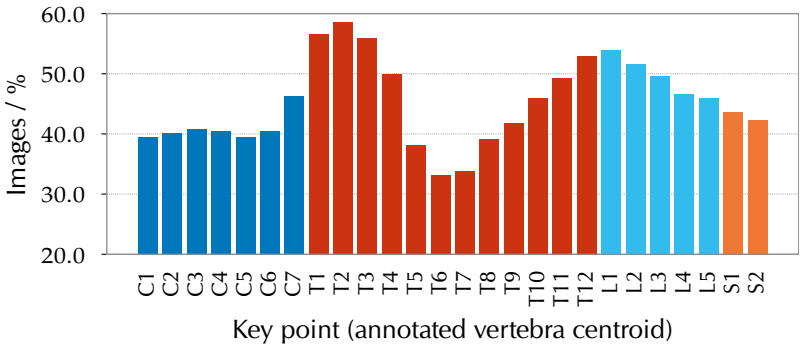


Figure 10.7: Illustration of the amount of cervical (■), thoracic (■), lumbar (■) and sacrum (■) vertebrae contained in all 302 images. Note the increased amount around T2 and L1 probably caused by overlapping section borders.

Thus, the task is the automatic detection (labeling) and localization of 26 vertebrae centroids. From top to bottom, they include 7 cervical vertebrae (labeled C1 to C7), 12 thoracic vertebrae (labeled T1 to T12), 5 lumbar vertebrae (labeled L1 to L5) and 2 additional vertebrae on the sacrum (labeled S1 and S2). The numbers of vertebrae contained in all 302 images per label is depicted in Fig. 10.7. They are distributed into 24.4 % cervical, 47.2 % thoracic, 21.1 % lumbar and 7.3 % sacrum vertebrae.

The key points per image vary from only 2 to all 26 vertebrae. The distribution w. r. t. the amount of vertebrae per image—representing the spine length per section—is shown in Fig. 10.8.

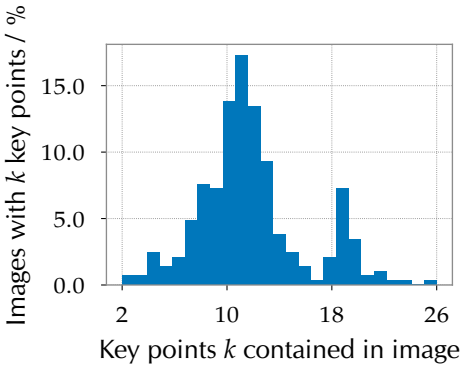


Figure 10.8: Distribution of the number of key points per image.

For the localization, Glocker et al. [78] proposed a localization threshold of 20 mm, which is used here as well if not stated otherwise.

Challenges

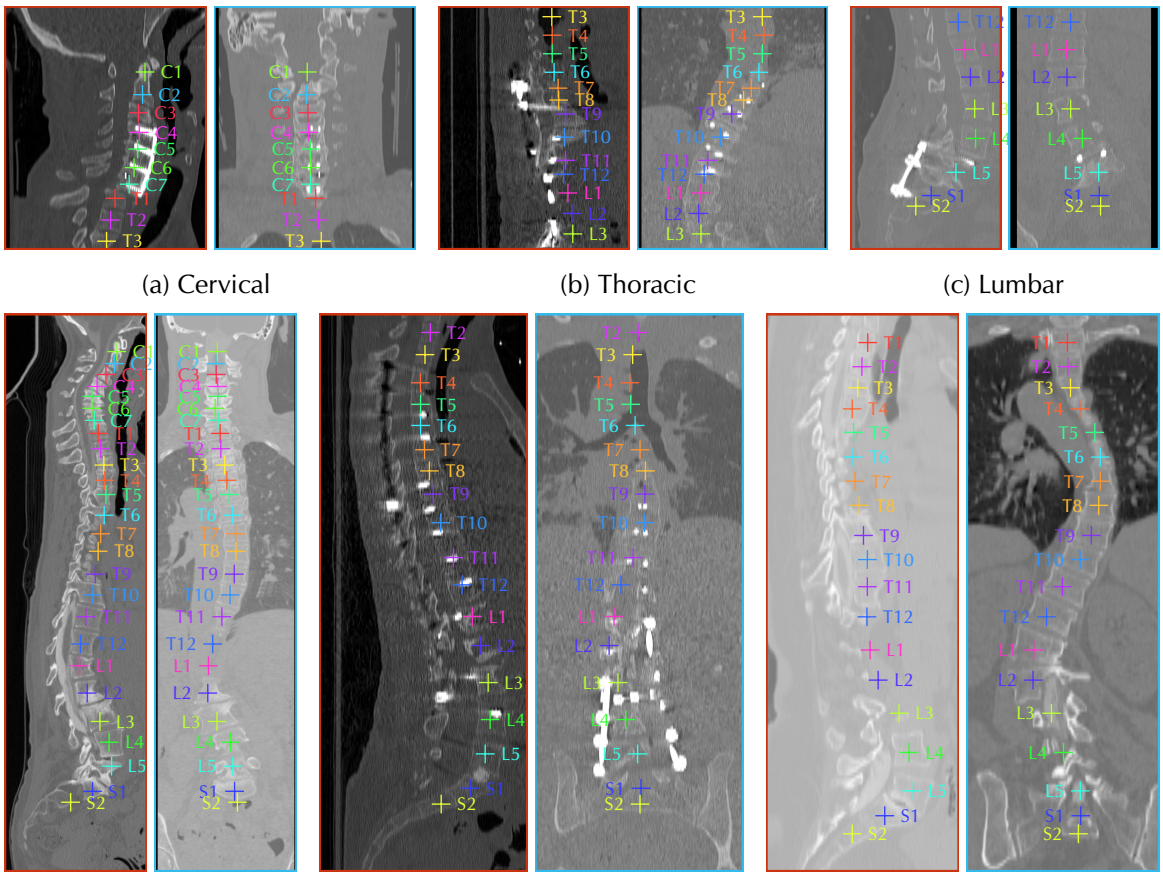
The main challenge of this dataset is the very repetitive nature of the vertebra bodies in combination with sections containing only few vertebrae. Especially the thoracic area is problematic w. r. t. differentiating the individual vertebrae, which is further complicated by the tight cylindrical mask around the spine effectively removing important discriminating structures.

As mentioned earlier, the images also show different kinds of pathologies like scoliosis and fractures as well as medical implants like fixations. Additionally, the image quality also varies drastically as well as the physical resolution. All these challenges render the task even harder.

Preparation

The dataset split of the MICCAI CSI 2014 challenge is used to ensure comparability, resulting in 242 training and 60 test images. To speed up the processing, the images have been resampled to an anisotropic resolution of $1 \times 1 \times 2.5\,\mathrm{mm}^3/\mathrm{vx}$.

From now on, this dataset is referred to as "spine sections". A few images from this dataset including the corresponding annotations are depicted in Fig. 10.9.



(a) Cervical (b) Thoracic (c) Lumbar

(d) Spanning all sections, first image shows all vertebrae

Figure 10.9: Illustration of example images taken from the spine sections dataset showing a (a) cervical, (b) thoracic and (c) lumbar section (with some overlap), and multiple (d) section-spanning images. The 3D volumes are illustrated as sagittal and coronal slices side-by-side while the slice index is computed as the median of all annotated slice indexes. The images are overlayed by the corresponding labeled (C1–C7, T1–T12, L1–L5 and S1–S2; color-coded "+") key point annotations. Beware that the annotations are potentially far away from the vertebrae centroids in slice normal direction. Note the various medical implants, the high-grade scoliosis as well as the bad image quality due to noise.

10.5 FULL SPINE CT SCANS

The fifth dataset consists of 157 in-house CT scans of the whole spine that were acquired at the Technical University of Munich. The 157 CT scans were acquired from 105 different patients, 36.2 % of which are females. At the start of the study, the patients had an average age of 67 years (41 to 88 years). Since some images were taken a few years later, consider this a lower bound on the patient age.

Anatomy

The images show thoracic and lumbar spine vertebrae, not including the head (i. e., no cervical vertebrae) but extending towards the tailbone (coccyx). While the images contain the whole body along the pos-

teroanterior axis, they are often (not always though) cropped around the spinal column along the left-right axis.

The images are sagittally sliced and have an average anisotropic resolution of $2.995 \times 0.783 \times 0.783\,\text{mm}^3/\text{vx}$ (illustrated as left-right ×posteroanterior ×superoinferior; ranging from a minimum of $2 \times 0.271 \times 0.271\,\text{mm}^3/\text{vx}$ to a maximum of $5 \times 1.512 \times 1.512\,\text{mm}^3/\text{vx}$) and an average size of $38 \times 529 \times 913$ vx (ranging from $20 \times 512 \times 512$ vx to $113 \times 1092 \times 2048$ vx), while covering an average physical extent of $112.1 \times 409.6 \times 590.7\,\text{mm}^3$ (ranging from $60 \times 139 \times 323\,\text{mm}^3$ to $339 \times 774 \times 774\,\text{mm}^3$)

Task

As outlined earlier, many diseases and pathologies affecting the spine have a high impact on life quality. Thus, an automated accurate and robust assessment of the spine is of great interest. This is even more true in trauma settings, where performance in terms of runtime is an additional driver. A common mode of operation is the segmentation of individual vertebrae, which allows for a further assessment of the individual vertebrae w. r. t. different pathologies like fractures. While recently model-free approaches such as semantic segmentation networks are being used for such tasks, it has been shown that the processi of the vertebrae still pose a problem for such methods [174]. An alternative is the model-based segmentation [63, 29], which efficiently uses prior knowledge to solve those issues. However, it requires multiple key point positions per vertebrae for an initial placement. Thus, the task is to localize those key points.

For each CT image, a semi-automatic approach was used to generate 102 key point positions, 6 for each of the 17 vertebrae T1 to L5. A model-based spine segmentation was performed as described in [105]. The outcome was manually checked for quality, discarding incorrect ones. The triangular mesh model for each vertebra consists of triangles labeled according to their anatomical position, including upper and lower end plate of the vertebral bodies, the foramen, processus spinosus and both processi transversus (see Fig. 10.10). Key point positions were calculated as center of mass of the respectively labeled triangles, resulting in 102 key point annotations per CT image (using an in-house dataset, since we are not aware of a public CT database with that many key point annotations). Although Glocker et al. [78] established a localization criterion w. r. t. the vertebrae already (see previous section),

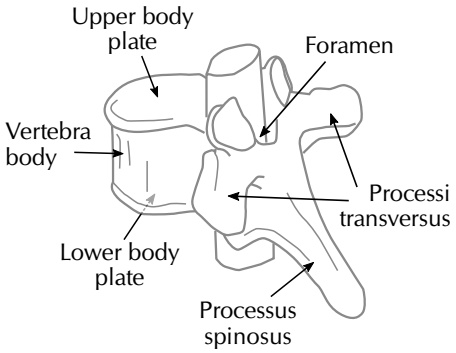


Figure 10.10: Illustration of the different parts of a vertebra for which key points were generated, based on [18, Fig. 7.23].

we use a tighter localization threshold of 10 mm due to problems with the aforementioned criterion explained later in Section 11.5.

Challenges

Similar to the previous dataset, a major problem of datasets featuring the spine is the repetition of locally ambiguous vertebrae, which often causes confusions, especially in the thoracic area. Unlike in the previous dataset, this is however lifted by the fact that large parts of the spine are visible in each scan. However, the number of key points was increased by a factor of ~ 5 from 26 to 102, most of which are not well-defined as center of mass positions of variable object structures, while the amount of data was reduced by a factor of ~ 2.

Furthermore, 36 of the 157 images (22.9 %) were labeled "incidental" (i. e., not healthy), potentially showing pathologies like fractures or implants.

Preparation

In order to follow best practices, the 157 CT scans were split into 5 patient-grouped folds in order to perform cross-validation. This resulted in, on average, 125 training images per cross-validation instance. To speed up the processing, the images have been resampled to an anisotropic resolution of $3\times1\times1\,\mathrm{mm}^3/\mathrm{vx}$.

From now on, this dataset is simply referred to as "full spines". Note that the scans often miss cervical vertebrae and that it is technically not the full spine, we just refer to it like this for simplicity since most of the vertebrae are involved. A few images from this dataset including the corresponding annotations and the localization criterion are depicted in Fig. 10.11.

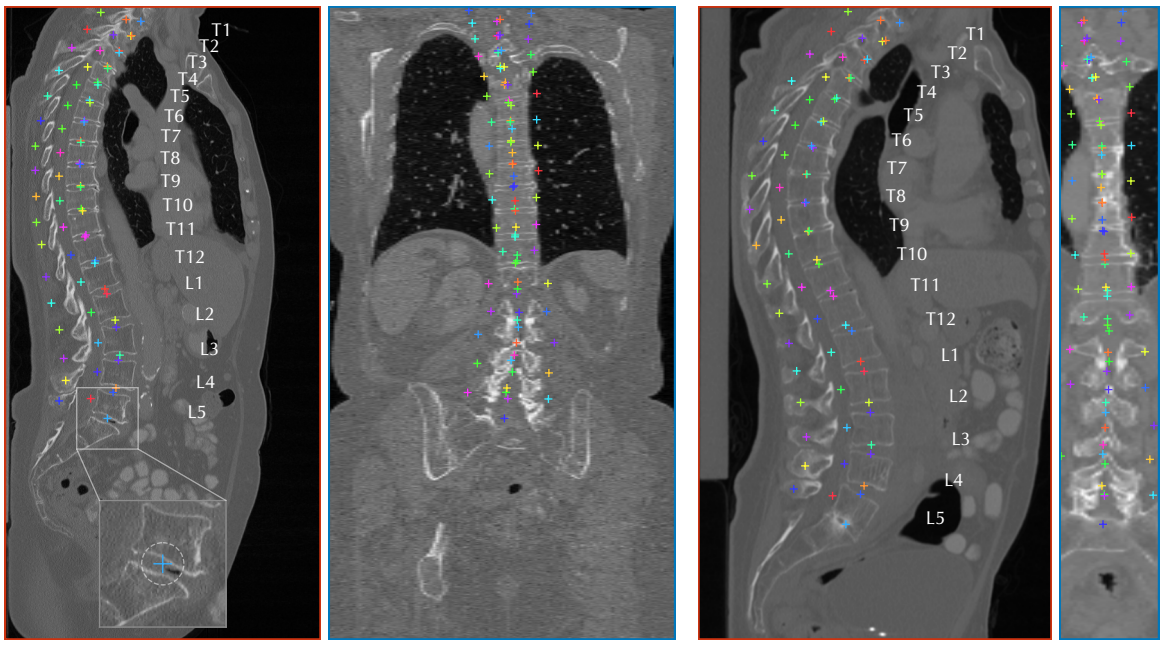


Figure 10.11: Illustration of example images taken from the full spines dataset. The 3D volumes are illustrated as sagittal and coronal slices side-by-side while the slice index is computed as the median of all annotated slice indexes. The images are overlayed by the corresponding key point annotations (color-coded "+"). Due to illustration purposes, only the vertebra levels are annotated with a label rather than the positions. Furthermore, the processi transversus are not shown in the sagittal slices and the processus spinosus is not shown in the coronal slices due to visiblity issues. Beware that the key point annotations are potentially far away from the actual position in slice normal direction. Note the fuzziness of the coronal slices in contrast to the sagittal slices. For more illustration of the same data see [132] and the corresponding supplementary material. The used localization criterion is illustrated in the inset in the first image.

10.6 LUMBAR SPINE MRI SCANS

This sixth dataset consists of 16 public 3D multi-modality MRI images, each showing at least 7 well-defined intervertebral discs (IVDs), that were collected from 8 patients in two time-spaced sessions and provided within the MICCAI 2018 IVDM3Seg challenge [215]. The images were generated using the Dixon protocol [57] with a 1.5 T Siemens MRI scanner, thus each 3D multi-modality MRI image is made of four channels each containing an aligned modality (i. e., fat, water, in-phase and opposed-phase).

In addition to these 16 released images, the organizers of the challenge kept 8 images from 4 patients unreleased for unbiased evaluation of submitted methods by the organizers. The following description is based on the public dataset that was released.

Anatomy

The images show the lower spine and thus mostly the lumbar vertebrae including the sacrum as well as some thoracic vertebrae, i. e., T10 to S2. Along the left-right axis, the images are tightly cropped around the spinal column, while they contain the whole body along the posteroanterior axis.

The images are sagittally sliced and have a constant anisotropic resolution of $2 \times 1.25 \times 1.25\,\mathrm{mm}^3/\mathrm{vx}$ (illustrated as left-right ×posteroanterior ×superoinferior) and a constant size of $36 \times 256 \times 256\,\mathrm{vx}$, while covering a physical extent of $72 \times 320 \times 320\,\mathrm{mm}^3$.

Task

Low back pain is still a dominant health problem in the population affecting general well-being and work ability. Furthermore, it is a major cause of disability. Various clinical studies (e. g., [88]) repeatedly reported a strong association between low back pain and the degeneration of the intervertebral discs (IVDs). Hence, the automatic assessment of the IVD tissue is of great interest especially in MRI, as it provides excellent soft tissue contrast.

The task is to automatically segment the 7 IVDs from T11 to S1 (i. e., the first IVD is between T11 and T12 while the seventh and last is between L5 and S1). For each of the seven IVDs, binary reference segmentations that were manually annotated are provided. We used the center of mass of each one to derive a target key point position for each IVD.

Challenges

Similar to the previous spine datasets, a major problem is the repetition of the locally ambiguous structures. This is even more true for IVDs, as they provide less distinguishable features than the vertebrae. In combination with the few amount of images, this is a severe problem for machine-learning-based approaches, as they are prone to over-fitting and often fail to generalize.

Another problem is the usage of MRI, as it provides unnormalized values that need to be normalized properly in order for a method to generalize from one scan to another. This is unlike CT, where the values are already normalized in the Hounsfield scale and are thus comparable across scanners.

Preparation

Since no patient identifiers were made available with the images, a standard 8-fold cross-validation based on patient splits (such that data from a given patient appears only in the training or in the test data) can not be performed. Instead, we decided to apply our method to every (out of $\binom{16}{2} = 120$) possible cross-validation configuration. These 120 configurations contain the 8 true patient splits and 112 configurations where the second image from the two test subjects is contained in the training set. This renders the results slightly over-optimistic. Nevertheless, we feel that this is a fair way to avoid a bias either towards a single test subject (with probability $^8/_{120} = 6.67\,\%$) or still using a non-correct patient split (with probability 93.3 %) when using a single, randomly selected cross-validation fold. We refer to this dataset as the "released lumbar spines", while we refer to the 8 held back images as "non-disclosed lumbar spines". Two images from this dataset including the corresponding annotated segmentations and derived key point positions are depicted in Fig. 10.12. Similarly to the previous dataset, we use a localization threshold of 10 mm.

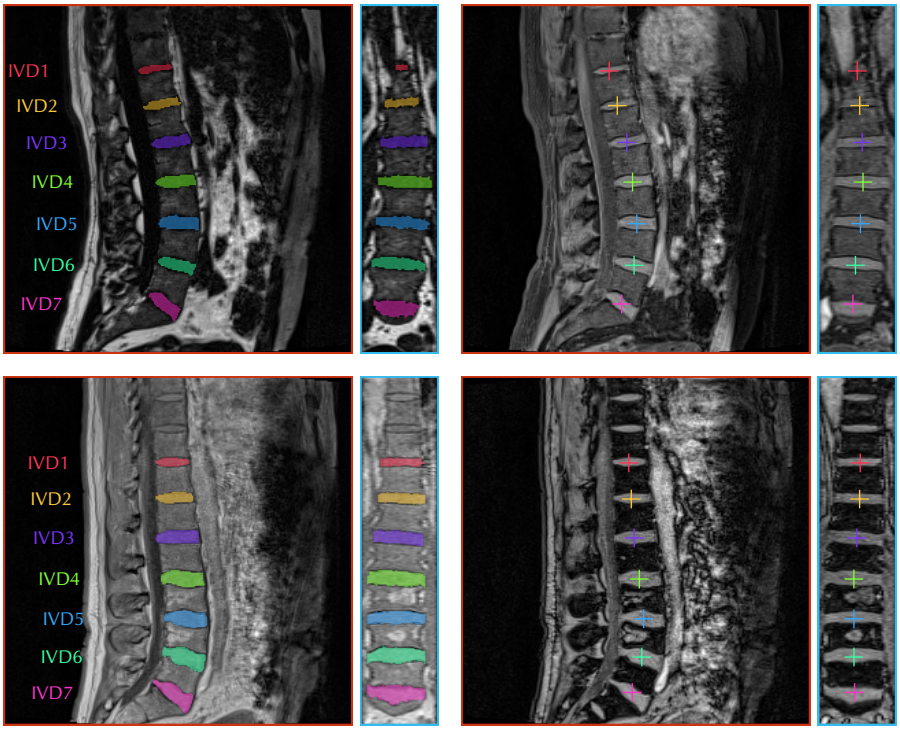


Figure 10.12: Illustration of two example images (one per row) taken from the lumbar spines dataset. The 3D volumes are illustrated as sagittal and coronal slices side-by-side and each pair showing a different MRI modalitiy (fat, water, in-phase and opposed-phase from left to right and top to bottom). The slice index is computed as the median of all annotated slice indexes. The left images are overlayed by the corresponding annotated segmentations (color-coded), while the right images are overlayed by the derived key point positions (color-coded "+"). Beware that the key point annotations are potentially far away from the actual position in slice normal direction.

10.7 OVERVIEW

A comparative overview of some key parameters of the six previously introduced and later used datasets is given in Table 10.1. For more details see the corresponding sections of Chapter 10.

Table 10.1: Listing of all used datasets and the important dataset parameters image modality, number of key points (N; minimal key points per image in parenthesis), number of training (K_{train}) and test (K_{test}) images and (average) image resolution (resampled resolution in parenthesis). A k-fold cross-validation is indicated by a times prefix of the number of images. Remember that 3D resolutions are illustrated as left-right×posteroanterior×superoinferior, while 2D resolutions are illustrated as left-right×posteroanterior. Average values are indicated by a "∼" prefix.

DATASET	MODALITY	KEY POINTS N	IMAGES $K_{\text{train}} + K_{\text{test}}$	RESOLUTION (OR SIZE)
Hands	X-ray	12	395 + 410	∼1212×2148 px* (∼347×600 px)
Legs	X-ray	(2–)6	5 × ∼(538 + 122)	0.143×0.143 mm²/px (1×1 mm²/px)
Chests	X-ray	16	3 × (428 + 214)	∼0.147×0.147 mm²/px (1×1 mm²/px)
Spine sections	CT	(2–)26	242 + 60	∼0.336×0.336×2.059 mm³/vx (1×1×2.5 mm³/vx)
Full spines	CT	102	5 × ∼(125 + 32)	∼2.995×0.783×0.783 mm³/vx (3×1×1 mm³/vx)
Lumbar spines	MRI	7	120 × (14 + 2) (+ 8)	2×1.25×1.25 mm³/vx (as original)

* This dataset does not provide the image resolution, thus the original average image size as well as the processed image size is stated.

11 METRICS

In order to quantitatively evaluate the performance of the method w. r. t. detection and localization, a few common metrics and a new joint metric is used. Each metric is exactly formulated in the following sections.

11.1 OUTCOME TYPES

Given that the detection and localization is a joint task, there exist multiple outcome types for a key point identified by the combination of the annotation and the respective prediction.

11.1.1 *Detection*

Each key point i is *annotated* as either missing ($\overset{\star}{s}_i = 0$), or as present ($\overset{\star}{s}_i = 1$), in which case the annotated position is given by $\overset{\star}{\boldsymbol{x}}_i \in \mathbb{R}^D$. Similarly, each key point i is *predicted* as either missing ($\hat{s}_i = 0$), or as present ($\hat{s}_i \geq 1$), in which case the predicted position is given by $\hat{\boldsymbol{x}}_i = \boldsymbol{x}_{i,\hat{s}_i}$. Thus, a result for the i-th key point w. r. t. the *detection* task can be classified as one of the following four cases:

FALSE NEGATIVE (FN) The annotation states that the key point is present, while the prediction states the key point as missing:

$$\overset{\star}{s}_i = 1 \quad \wedge \quad \hat{s}_i = 0.$$

FALSE POSITIVE (FP) The annotation states the key point as missing, while the prediction states the key point as present:

$$\overset{\star}{s}_i = 0 \quad \wedge \quad \hat{s}_i \geq 1.$$

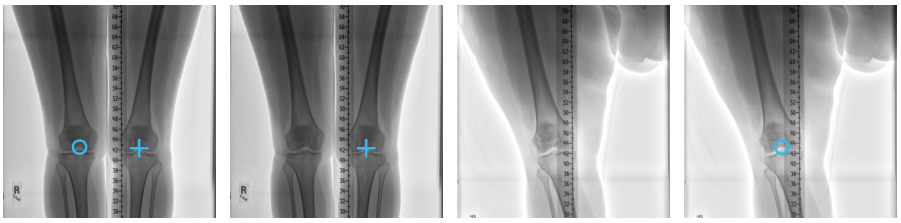
TRUE NEGATIVE (TN) The annotation as well as the prediction state the key point as missing:

$$\overset{\star}{s}_i = 0 \quad \wedge \quad \hat{s}_i = 0.$$

TRUE POSITIVE (TP) The annotation as well as the prediction state the key point as present:

$$\overset{\star}{s}_i = 1 \quad \wedge \quad \hat{s}_i \geq 1.$$

These four cases are visually depicted in Fig. 11.1.



(a) TP (b) FN (c) TN (d) FP

Figure 11.1: Illustration of the four possible outcome types w. r. t. the detection task, assuming that the target key point is the left knee joint (on the right side in the image) as in the legs dataset (see Section 10.2). The first two cases (a) TP and (b) FN are applicable when the target key point is visible in the image, the position of which is marked by a "+". A corresponding predicted position (i. e., $\hat{s}_i \geq 1$ selecting $\boldsymbol{x}_{i,\hat{s}_i}$) is marked by a circle. The second two cases (c) TN and (d) FP are applicable when the target key point is not present, as caused in this case by an amputation. Note that the location of the predicted position may be arbitrary in the (a) TP case and needs further evaluation.

11.1.2 *Localization*

By further assessing the TP case, we can evaluate the outcome w. r. t. the *localization* task. To do so, the TP case is split into two sub cases using a localization criterion $c\colon \mathbb{R}^D \times \mathbb{R}^D \to \{\text{true}, \text{false}\}$ to decide whether the i-th key point is correctly localized by comparing the annotated key point position $\dot{\boldsymbol{x}}_i$ to the predicted key point position $\hat{\boldsymbol{x}}_i$:

LOCALIZED TRUE POSITIVE (L-TP) The annotation as well as the prediction state the key point as present, and the predicted key point position is considered as correctly localized w. r. t. the annotated key point position:

$$\dot{s}_i = 1 \quad \wedge \quad \hat{s}_i \geq 1 \quad \wedge \quad c(\dot{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_i).$$

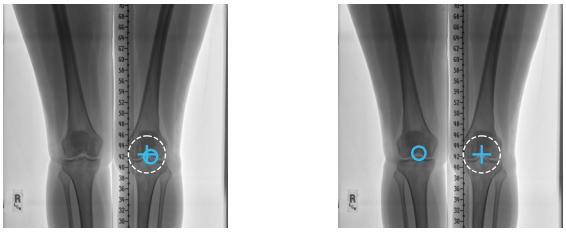
MIS-LOCALIZED TRUE POSITIVE (M-TP) The annotation as well as the prediction state the key point as present, and the predicted key point position is considered as not correctly localized w. r. t. the annotated key point position:

$$\dot{s}_i = 1 \quad \wedge \quad \hat{s}_i \geq 1 \quad \wedge \quad \neg c(\dot{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_i).$$

Commonly, the Euclidean distance between the predicted position and the annotated position is compared to a localization threshold R to decide whether a key point is localized correctly. Recall that we used R before to derive, e. g., the target Gaussian distribution in the RTE training scheme in Section 5.1.2, which is a reasonable parameter choice as it relates the training to the localization criterion. This criterion can be formalized as

$$c(\dot{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_i) = \left(\|\dot{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i\|_2 < R \right) \tag{11.1}$$

If not stated otherwise, this is the default localization criterion applied (see, e. g., Section 10.3 for a different type of localization criterion). The distinction is visualized in Fig. 11.2. Note that this distinction can also be handled differently. For instance, M-TP cases are treated as outliers in [151].



(a) L-TP (b) M-TP

Figure 11.2: Illustration of the two possible localization outcome types (a) L-TP and (b) M-TP assuming that the target key point is the left knee as in the legs dataset (see Section 10.2) with a "+" marking the annotated position and a circle marking the corresponding predicted position. A simple Euclidean distance threshold localization criterion as in Eq. (11.1) is assumed, the threshold R is depicted as large dashed circle. Note that the threshold is in practice much lower.

To this end, the prediction result of the i-th key point in the k-th image is given by $o_i^k \in \{\text{FN}, \text{FP}, \text{TN}\} \cup \text{TP} = \{\text{FN}, \text{FP}, \text{TN}, \text{M-TP}, \text{L-TP}\}$

11.2 SUCCESS RATE

In order to jointly quantify the performance of the detection as well as the localization task, the *success rate* is introduced. The success rate computes the fraction of correct prediction results, i. e., TN and L-TP cases, over a set of predictions and is reported in percent (%). Note that L-TP is considered instead of TP to combine the detection and the localization task by counting cases that have been correctly identified as missing or that have been correctly identified as present and have been correctly localized w. r. t. some localization criterion.

Using the Iverson bracket notation $[\text{S}] \colon \{\text{false}, \text{true}\} \to \{0, 1\}$ to yield 1 if the statement S is true and 0 if it is false, the success rate on key point level for a set of $K = |\mathcal{D}|$ images is defined as

$$\frac{1}{KN} \cdot \sum_{k=1}^{K} \sum_{i=1}^{N} \left[o_i^k \in \{\text{TN}, \text{L-TP}\} \right]. \tag{11.2}$$

The success rate can also be computed on image level, i. e., quantifying the number of images where all key points are correct, and is given by

$$\frac{1}{K} \cdot \sum_{k=1}^{K} \prod_{i=1}^{N} \left[o_i^k \in \{\text{TN}, \text{L-TP}\} \right]. \tag{11.3}$$

Note that both of these metrics can also be computed for subsets of the used key points.

11.3 LOCALIZATION RATE

The performance of the localization task can individually be assessed by evaluating only TP cases—i. e., L-TP and M-TP—and counting all correctly localized L-TP cases. Following the same notation as before, the localization rate on key point level is defined as

$$\frac{1}{|\text{TP}|} \cdot \sum_{k=1}^{K} \sum_{i=1}^{N} \left[o_i^k = \text{L-TP} \right], \tag{11.4}$$

with

$$|\text{TP}| = \sum_{k=1}^{K} \sum_{i=1}^{N} \left[o_i^k \in \{\text{M-TP}, \text{L-TP}\} \right] \tag{11.5}$$

being the amount of TP cases. This localization rate can also be computed on image level, i. e., quantifying the amount of images where all TP key points where localized correctly, and is formally given by

$$\frac{1}{K} \cdot \sum_{k=1}^{K} \prod_{i=1}^{N} \left[o_i^k \in \{\text{L-TP}, \text{FP}, \text{FN}, \text{TN}\} \right]. \tag{11.6}$$

Note that the localization rate is equivalent to the success rate in case the dataset does not pose the detection problem. As before, it might also be computed for a subset of key points rather than all key points.

11.4 LOCALIZATION ERROR

The performance of the localization task (i. e., TP cases only) is further quantified in terms of the localization error, which corresponds to the Euclidean distance

$$\text{dist}(\overset{\star}{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_i) = \|\overset{\star}{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i\|_2 \tag{11.7}$$

between the annotated key point position and corresponding predicted one (see Fig. 11.3). It is often reported in terms of the average

$$\frac{1}{|\text{TP}|} \cdot \sum_{k=1}^{K} \sum_{i=1}^{N} \left[o_i^k \in \{\text{M-TP}, \text{L-TP}\} \right] \cdot \text{dist}(\overset{\star}{\boldsymbol{x}}_i^k, \hat{\boldsymbol{x}}_i^k) \tag{11.8}$$

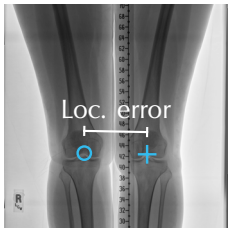


Figure 11.3: Illustration of the localization error (Euclidean distance) as computed for an annotated position ("+") and a corresponding predicted position (circle).

computed over all TP cases over a set of images. However, different statistics like standard deviation or different subsets of key points may be used as well when aggregating the individual localization errors from Eq. (11.7).

Recently, it has been observed that other groups (e. g., [215] and [174]) introduce an artificial, very large penalty value as the localization error for cases not classified as TP. However, this effectively skews the localization error up to the point of no reasonable interpretability. This is why—in this thesis—the localization error is instead computed on the subset of TP key point predictions and is reported in combination with the classification distribution (if necessary).

The localization error is reported as distance in mm if the dataset at hand has an accompanying spatial resolution. If no spatial resolution is available, the localization error is reported in px / vx.

11.5 IDENTIFICATION RATE

For the evaluation of the prediction results on the public spine sections benchmark dataset, Glocker et al. [77] established the *identification rate* as an additional metric. The identification rate classifies TP predictions into correctly and incorrectly identified predictions and reports the fraction of correctly identified predictions over all TP predictions in percent (%).[1]

A prediction is treated as correctly identified if (1) the corresponding localization error is below 20 mm and (2) the closest annotated key point is the predicted key point (identical label):

$$\operatorname{dist}(\hat{\boldsymbol{x}}_i^\star, \hat{\boldsymbol{x}}_i) < 20\,\mathrm{mm} \quad \wedge \quad i = \arg\min_{i'} \operatorname{dist}(\hat{\boldsymbol{x}}_{i'}^\star, \hat{\boldsymbol{x}}_i). \tag{11.9}$$

The identification rate is only reported for results achieved on the spine sections dataset in order to compare with other methods, but also due to two problems rendering this measure suboptimal. First, as also acknowledged in [175] and [174], stating just localization metrics in combination with a high identification rate may disguise an insufficient detection rate. Second, the condition in Eq. (11.9) is too relaxed. It tries to handle proper spine vertebrae separation, but does not account for the intervertebral discs and small vertebrae as in the cervical area due to a generally too large localization threshold of 20 mm. The problem is illustrated in Fig. 11.4. Note that the success rate with a properly chosen localization threshold R solves both of those problems.

[1] This has been verified by checking the originally used Matlab code, which has been kindly provided by Ben Glocker via e-mail.

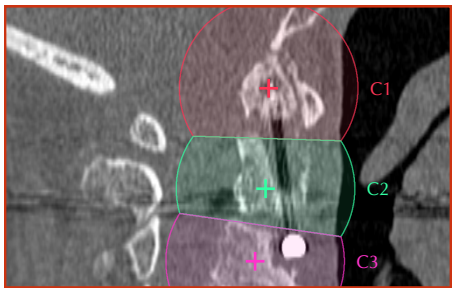


Figure 11.4: Illustration of the identification criterion as given in Eq. (11.9) for a CT scan from the spine sections dataset containing the three cervical vertebrae C1, C2 and C3 (top to bottom). Depicted is just one sagittal slice of the volume from the average annotated key point position, visualizing the criterion in form of key-point-specific (color-coded) surrounded areas where a corresponding predicted key point is treated as correctly identified. Note that the criterion clearly allows predictions to not be near the annotated position or even within the respective vertebra.

Part IV
EVALUATION

12

GENERAL APPLICABILITY AND EASY TRANSFERABILITY

As explained earlier, a major problem of graph-based localization approaches is the ad-hoc nature which we tried to tackle using a topology optimizing training algorithm. The assumed key benefit is the easy transferability of the approach to different datasets—involving different image modalities, dimensionalities, target structures and amounts of key points—and thus the general applicability to different medical localization targets. In the following, we try to empirically verify this claim.

12.1 EXPERIMENTAL SETUP

To do so, we apply the method to the different datasets with different numbers of key points and different corresponding target objects. The method uses the "default" configuration, i. e., one RTE appearance model (see Section 5.1) per key point represented as unary potentials in combination with all three binary (i. e., pairwise) potential functions—utilizing distance, angle and vector; see Section 6.1—per key point pair. The idea behind this configuration is to provide a quickly trainable and testable base configuration yielding decent performance in various medical applications requiring the adjustment of only few parameters. Furthermore, it should provide a baseline for dataset-specific tuning and illustrates the *pool of potential functions* idea, as it is clear that not all potential functions are useful and proper selection of those should yield better performance.

12.1.1 *Data*

The four datasets hands, legs, chests and spine sections have been used to cover different image dimensions, modalities and target objects. We use the dataset splits proposed in the literature (if available, otherwise, k-fold patient-grouped cross-validation is performed) and the corresponding localization criterions (see the dataset-specific sections in Chapter 10). An overview of the parameters of the respective datasets and references to the respective descriptive sections is given in Table 10.1. Note that the other two datasets also feature the spine and are thus not included in this experiment, but rather in later ones evaluating different properties.

For each dataset, the set of K_{train} training images $\mathcal{D}_{\text{train}}$ is further divided into three non-overlapping subsets $\mathcal{D}_{\text{pots}}$, $\mathcal{D}_{\text{graph}}$ and $\mathcal{D}_{\text{val}}$, containing 30 %, 60 % and 10 % of randomly selected training images (patient-grouped), respectively. The purpose of these datasets is explained in the following subsections. The set of test images $\mathcal{D}_{\text{test}}$ is not changed and used as is.

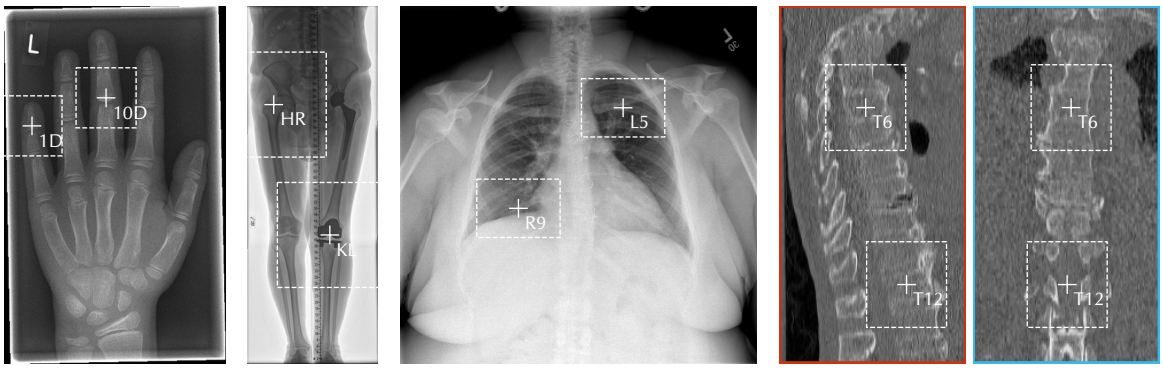
12.1.2 *Parameters*

The first training data subset $\mathcal{D}_{\text{pots}}$ is used to train the individual potential functions, which corresponds to training the local appearance models in form of RTEs and estimating the parameters of the binary spatial statistics. While the latter does not have any meta parameters, the former requires only a few parameters. The patch size $\boldsymbol{A}_i$ is estimated by inspecting the target anatomy at hand, visually illustrated in Fig. 12.1. The number of features V_i and trees T_i default to 128 and 96, respectively, heuristically estimated in initial experiments. We do increase those values for the chests and spine sections dataset due to a harder detection caused by reduced abdominal contrast and the added third dimension, respectively. An overview of the RTE parameters is shown in Table 12.1.

Table 12.1: Listing of the used RTE parameters patch size $\boldsymbol{A}_i$, number of features V_i and number of trees T_i for each of the four datasets that have been used in this experiment.

DATASET	PATCH SIZE $\boldsymbol{A}_i$	FEATURES V_i	TREES T_i
Hands	101×101 px	128	96
Legs	351×351 px	128	96
Chests	100×70 px	256	144
Spine sections	70×70×30 vx	256	144

The second training data subset $\mathcal{D}_{\text{graph}}$ is used to learn the potential weights $\boldsymbol{\Lambda}$ and the "missing" energies $\boldsymbol{B}$ (see Chapter 7). Optimizing the weights and energies on data independent of $\mathcal{D}_{\text{pots}}$ greatly improves generalization, since the potential energies are not computed on images the potential functions were trained on. The last training data subset $\mathcal{D}_{\text{val}}$ is used as validation set in order to perform model selection w. r. t. different meta parameters used during the optimization of the potential weights and "missing" energies. A grid search is performed over the regularization type and the regularization weight, the energy margin and the learning rate for the values listed in Table 12.2. The model that achieved the highest success rate computed on $\mathcal{D}_{\text{val}}$ is selected for final evaluation on $\mathcal{D}_{\text{test}}$. Note that this is done multiple times for the chests and legs datasets (cross-validation), while only one time for the hands and spine sections datasets (pre-defined training test split).



(a) Hands (b) Legs (c) Chests (d) Spine sections

Figure 12.1: Illustration of the RTE patch sizes listed in Table 12.1 for the four different datasets (a) hands, (b) legs, (c) chests and (d) spine sections. For each dataset, two annotated key points are marked (white "+") and the corresponding patch size is illustrated as rectangle centered around it. A mid-sagittal and mid-frontal slice is shown for the (d) spine sections, respectively.

Table 12.2: Listing of the optimization parameters (see Chapter 7) and corresponding values that form the combinatorial grid search space containing 204 parameter sets to evaluate.

PARAMETER	VALUES
Regularization $\Omega(\boldsymbol{\Lambda})$	0, $\|\boldsymbol{\Lambda}\|_1$, $\|\boldsymbol{\Lambda}\|_2$
Regularization weight θ	0.001, 0.004, 0.016, 0.064, 0.256, 1.024, 4.096, 16.384
Margin m	0.1, 0.5, 5, 10
Learning rate η	0.01, 0.05, 0.1

12.2 DETECTION AND LOCALIZATION PERFORMANCE

The finally achieved detection and localization performance is listed in Table 12.3. The subjectively perceived severity of the dataset w. r. t. the detection and localization task is reflected by the achieved results. For instance, the arguably most easy hands dataset has a high success rate of 99.7 %, where as the most difficult spine sections dataset has a success rate of 87.2 % (with respect to the corrected test set, see later Section 12.5.4). Note that this fairly "default" configuration achieved decent results on all four different datasets, featuring different numbers of key points, target structures and image modalities, without making any assumptions about the dataset and the target objects except for the patch size. It provides a good starting point for dataset and task-specific optimizations.

In the following, we discuss the results obtained for each dataset individually in more detail while focusing on the dominant error types. Some example images including the predicted results are shown in Fig. 12.2 for each dataset.

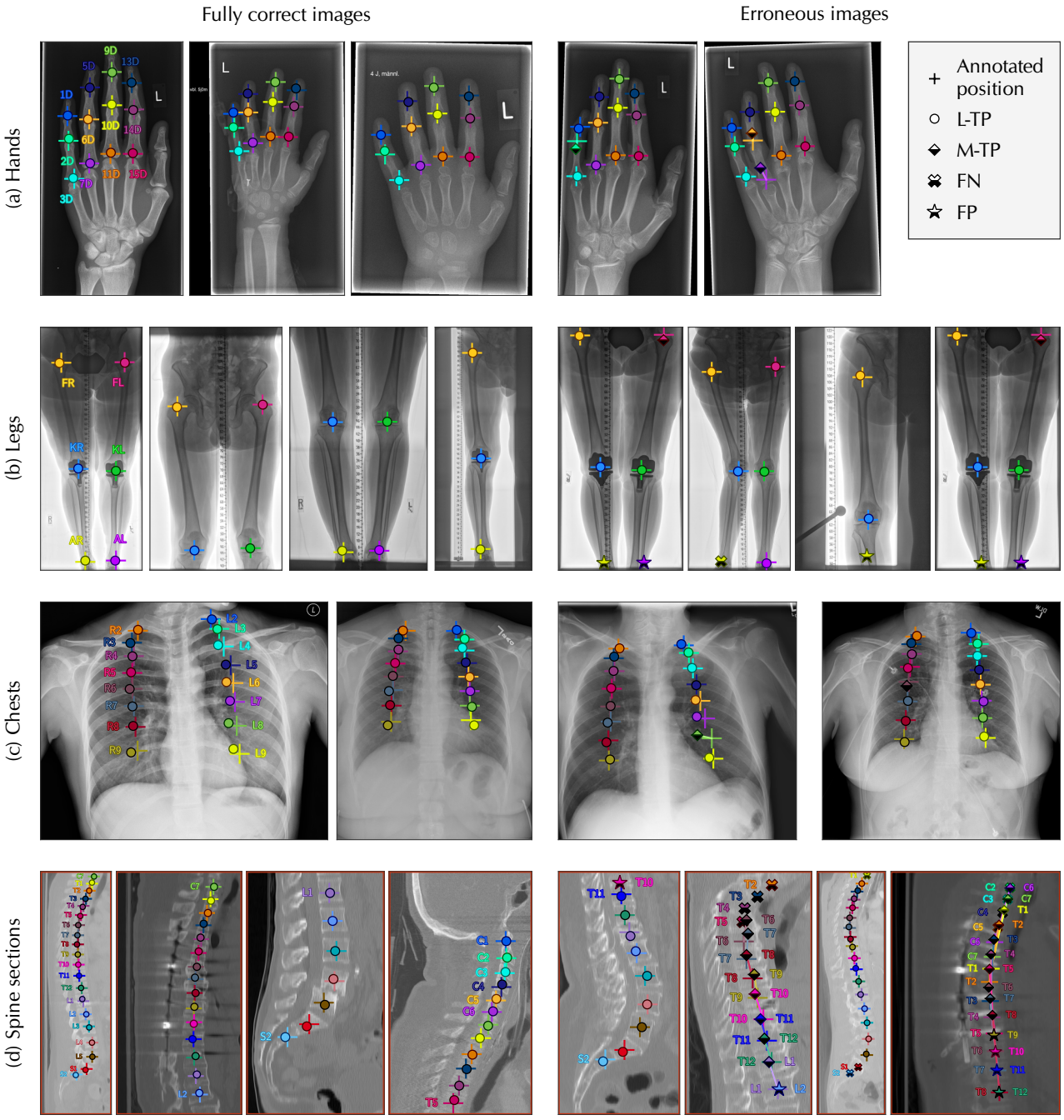


Figure 12.2: Illustration of fully correct (left column) and partly correct (right column) test results achieved on the (a) hands, (b) legs, (c) chests and (d) spine sections datasets spread across the four rows, respectively. For each key point, the annotated and predicted position are color-coded and the marker type of the predicted position indicates the outcome type. The right column illustrates all possible error cases, i. e., mis-localizations (L-TP), missing detection (FN) and incorrect detection (FP). For simultaneous visibility of all vertebrae in the (d) spine sections, we show averaged slices of the sagittal plane only. We can see two cases of shifted segments where multiple vertebrae are confused and the confusion is propagated. Recently, it was found (see text later) that some test images including the last illustrated one were incorrectly annotated, which renders the last image actually fully correct w. r. t. the corrected test image (see Section 12.5.4). Note the accurate localization despite surgical implants and fractures.

Table 12.3: Final results achieved by our detection and localization approach after CRF optimization (starting from a fully connected and fully loaded graph) in terms of success rate in percent (on key point as well as image level, i. e., all key points are correct), localization rate in percent and the average localization error.

DATASET	SUCCESS RATE / %		LOC. RATE / %	AVG. LOC. ERROR
	KEY POINTS	IMAGES		
Hands	99.7	98.0	99.7	1.5 px
Legs	94.3	77.0	95.1	3.6 mm
Chests	92.2	72.0	92.2	5.6 mm
Spine sections	87.2	33.3	83.1	8.8 mm

12.2.1 *Hands*

An average success rate of 99.7 % over all key points has been achieved on the hands dataset. Of the 410 test images, 98.0 % were fully correct, i. e., all 12 key points were localized correctly, while 4, 2, 1 and 1 image still contained 1, 2, 3 and 5 mis-localized key points. The failed images mostly contained anatomical abnormalities (i. e., strong deviations from the mean hand). The localization error in terms of Euclidean distance between the annotated position and the predicted position is (on average) 1.5 px.

12.2.2 *Legs*

An average success rate of 94.3 % over all key points has been achieved on the legs dataset. The overall success rate as well as the key-point-specific success rates, averaged over both legs, are indicated by TN + L-TP in Fig. 12.3, where the outcome type distribution is visualized. Looking at that chart, we see that the detection task (but not necessarily the localization task) was solved on average in 84.4 % L-TP + 9.8 % TN + 4.4 % M-TP = 98.6 % of the cases. The largest amount of errors is due to mis-localization with 4.4 %, in contrast to incorrect detection with only 0.5 % FP + 0.8 % FN = 1.4 %. The comparably worse success rate (TN + L-TP) for the hip (93.8 %) and ankle (90.6 %) in contrast to the knee (98.4 %) is related to a generally worse performance of the localizer close to the image boundaries. The localization error is (on average) 3.6 mm. Furthermore, the approach was able to correctly handle all key points in 77.0 % of the test images.

12.2.3 *Chests*

An average success rate of 92.2 % over all key points has been achieved on the chests dataset. This corresponds to 72.0 % fully correct images.

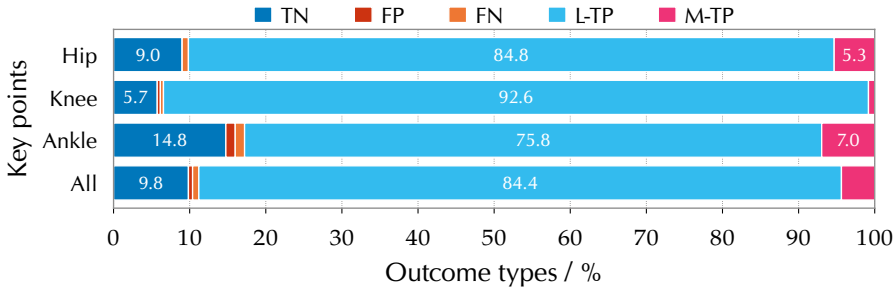


Figure 12.3: Distribution of the five outcome types for the hip (HL + HR), knee (KL + KR), ankle (AL + AR) and all key points of the legs dataset.

Many of the errors are caused by insufficient sets of localization hypotheses (i. e., no good candidate) for the posterior ribs near the abdomen (8 to 9), which is mostly caused by the very low contrast in that area. An average localization error w. r. t. the annotated key point position of 5.6 mm has been achieved. However, computing the localization error w. r. t. the annotated centerline seems more reasonable, given that the localization criterion tolerates shifts along the centerline. As expected, with a value of 2.6 mm it is lower than the original value.

12.2.4 *Spine sections*

An average success rate of 87.2 % over all key points has been achieved on the corrected spine sections test set (see Section 12.5.4). Error rates for mis-detection (FP + FN; 6.0 %) and mis-localization (M-TP; 6.9 %) are rather similar (see Fig. 12.4). In general, the performance for the cervical vertebrae is better than for the thoracic and lumbar vertebrae, which seems reasonable given that the thoracic and lumbar vertebrae provide a less distinctive shape compared to the cervical ones. Especially in spine sections containing only thoracic vertebrae, shifts along the spine (i. e., mis-labeling by one or more vertebrae along the spinal chain) are a typical source of error. In 33.3 % of the images, all 26 landmarks were handled correctly and an average (with standard deviation) localization error of (8.8 ± 13.4) mm has been achieved.

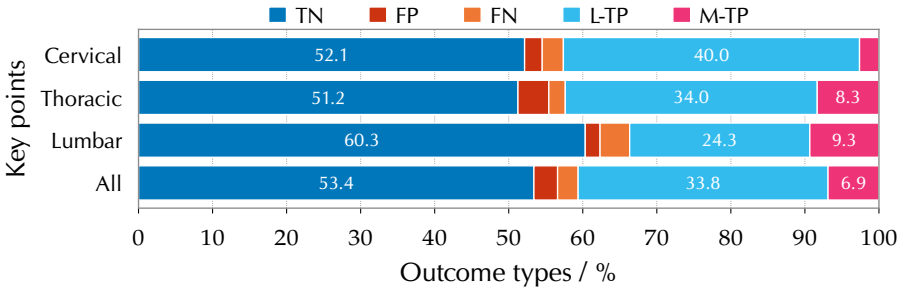


Figure 12.4: Distribution of the five outcome types for the cervical (C1–C7), thoracic (T1–T12), lumbar (L1–L5 + S1 + S2) and all key points of the spine sections datasets.

12.3 RESULT OF THE OPTIMIZATION

The necessity of the CRF optimization is visualized in Fig. 12.5 by comparing the performance in terms of success rate before and after the optimization and in addition to only using the RTE localizers (if applicable) without the CRF. Note that the optimization has been carried out until no improvement (computed on the validation set) has been observed for a prolonged period of time ($\sim 30\,\text{min}$), finally using the parameters with the highest success rate achieved on the validation set. As can be seen, the optimization improves the performance—as expected—in contrast to an unoptimized CRF. In case of the legs and spine sections datasets, this is mostly attributed to the proper estimation of the "missing" energies, which is evident when comparing the drop in performance of using an unoptimized CRF on the legs dataset—requiring the "missing" label—and the non-existing drop in the hands dataset, which does not use the "missing" label. For the hands and chests datasets, solely the potential weights cause the improvement.

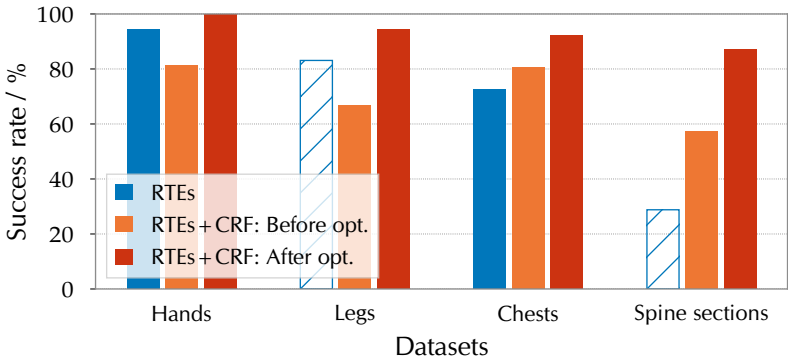


Figure 12.5: Comparison of the success rate achieved just by using the RTEs (■), an unoptimized CRF (iteration 0 of the optimization; ■) and after optimization (■) for the four different datasets. In case of the legs and spine sections datasets, just using the localizer is biased towards an incorrect penalization of FP cases, hence these results are depicted with less emphasis and should not be compared.

Table 12.4: Listing of the selected optimization parameter values (see Chapter 7) that were found by performing a grid search over validation data.

DATASET	META PARAMETERS			
	REG. $\Omega(\boldsymbol{\Lambda})$	REG. WEIGHT θ	MARGIN m	LEARN. RATE η
Hands	$\|\boldsymbol{\Lambda}\|_1$	0.0001	5	0.1
Legs	$\|\boldsymbol{\Lambda}\|_2$	0.0001	0.1	0.01
Chests	0	-	10	0.05
Spine sections	$\|\boldsymbol{\Lambda}\|_1$	0.0001	5	0.05

The finally chosen meta parameters for each dataset are listed in Table 12.4. Except for the strength of the regularization weight θ, there was no clear indication whether a specific parameter causes an isolated severe effect on the success rate in the tested value ranges, i. e., no strong systematic change in performance. In contrast, the weight regularization and the corresponding regularization weight θ can have a severe impact on the performance in terms of the success rate. Using either L^1 or L^2 penalization in combination with a small $\theta \leq 0.0001$ lead to improved results. However, if θ is too large, the performance in terms of success rate drops significantly. In Table 12.5, the best success rates that were achieved using the two types of regularization method are listed for the four datasets next to using no weight regularization. The results generally support that using some form of regularization is beneficial.

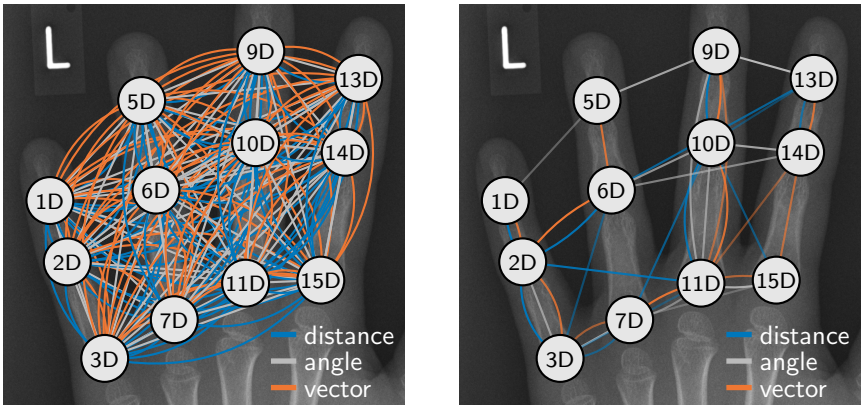
Table 12.5: Listing of the success rates (%) for the four different datasets in dependence of a different type of potential weight regularization.

REGULARIZATION	SUCCESS RATE / %			
	HANDS	LEGS	CHESTS	SPINE SECTIONS
None	99.3	93.9	**92.2**	85.2
L^1	**99.7**	94.0	91.3	**87.2**
L^2	99.6	**94.3**	90.4	85.3

12.4 DEDUCED FACTOR GRAPHS

Using a model-based approach provides the benefit of reasoning about the resulting model after optimizing it with data and possibly gaining some new insights. In this case, the evaluation of the different graph topologies deduced by the optimization (factor graphs), are of interest as they describe the correlation between the different key points. Thus, it has been tried to visually depict the resulting factor graphs in a reasonable way.

A simplified factor graph notation is visualized for the hands dataset before and after the CRF optimization in Fig. 12.6. The graph before the optimization describes the fully loaded and fully connected starting point of the optimization (just depicted for better comparison with the resulting topology after optimization). Note that no unary potential was dropped; thus, the unary potentials are not visualized in Fig. 12.6. It is evident that many potential functions were dropped, of the initially $210 = \binom{12}{2} \cdot 3 + 12$ potential functions 158 (75.2 %) were removed. Furthermore, it seems that a local connectivity pattern is preferred over long-range dependencies, as nearly all of them were removed. Interestingly, the variance in the finger placement (closed versus spread) is reflected in the choice of potential function near the finger tips, since for all pairs between them only the angle potential remained which is mostly unaffected by it. This is in contrast to, e. g., the distance potential in which case the Gaussian assumption might not be correct and consequently the potential has been correctly removed.

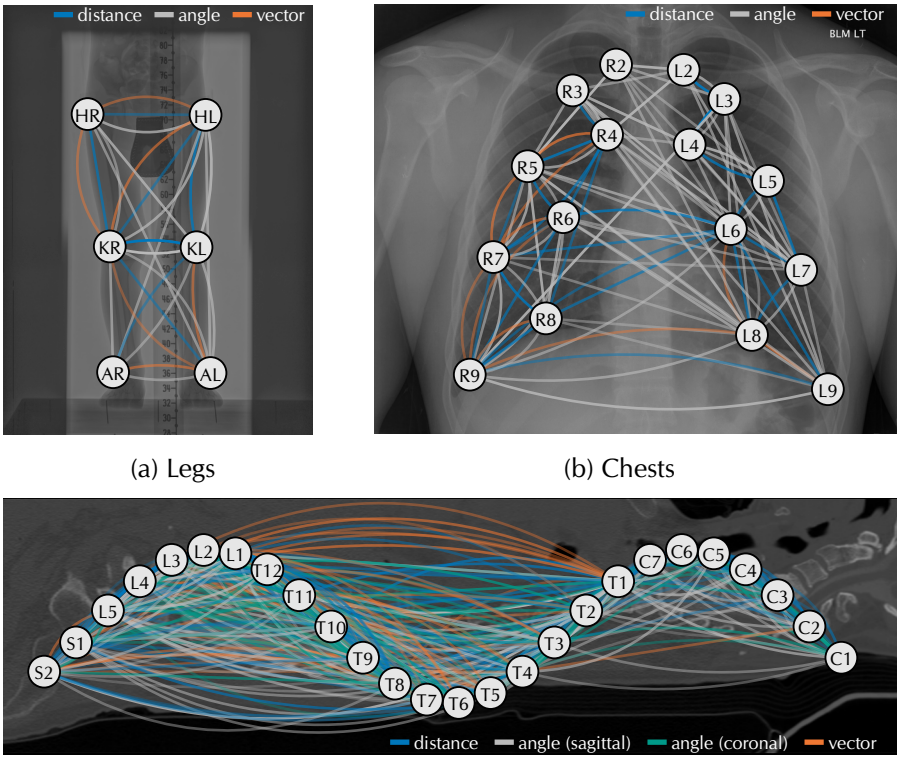


(a) Before optimization (b) After optimization

Figure 12.6: Illustration of the factor graph (a) before and (b) after the CRF optimization on the hands dataset. Due to visualization reasons, a simplified notation is used: Unary potential functions are excluded (none were dropped during the optimization) and binary potentials are simply indicated by edges while the potential function types are color-coded (i. e., ■ distance, ■ angle and ■ vector). The potential weight is manifested in the transparency of the edge, using a power transform of ¼ to map the weight into the transparency range of 0.2 to 1 to ensure visibility of all non-zero-weighted potentials. Note that the placement of the nodes in relation to the underlying image is just for visualization purposes.

The resulting graph topologies for the other three datasets are depicted in Fig. 12.7. It is obvious that the fully connected structure was not maintained for any graph. As before, many long range dependencies have been removed in favor of connections with a close spatial proximity. This effect, however, is less pronounced with increasing variability of the spatial constellation. For instance, the legs dataset has a

reasonably consistent spatial regime (except obviously for images with a small field of view) caused by the resampling to a constant height. In contrast to the spine sections dataset, where the spinal chord has more degrees of freedom w. r. t. the spatial variability. Thus, the latter needs more information which manifests in more long-range dependencies than used by the datasets before. In case of the legs and chests dataset, multiple graphs were estimated due to the cross-validation setup. While the graphs also showed the same general pattern of removed long-range dependencies in favor of short-range dependencies, they did not show resemblance in terms of the remaining types of potential functions and the connections in detail. Hence, the resulting graph topology is not only sensitive to the optimization, but also to the dataset distribution. This extends to the different folds of the legs and chests dataset, which results in different graph topologies and thus potentially different interpretations.



(a) Legs

(b) Chests

(c) Spine sections

Figure 12.7: Illustration of the resulting factor graphs after the CRF optimization on the (a) legs, (b) chests and (c) spine sections using the same visualization protocol as in Fig. 12.6. In case of the (a) legs and (b) chests dataset, the result of just one fold is depicted. Additionally, the (c) spine sections dataset has one angle for the sagittal plane (■) and one for the coronal plane (■) per key point pair. Note that the node positions are independent of the depicted image and re-arranged for better visibility. For a different type of illustration see [134].

The sparsification of potential functions and thus the resulting factor graph can majorly be affected by the choice of weight regularization. Thus, the amount of removed potential functions is compared for different regularization types next to the achieved success rates; the results of which are listed in Table 12.6. Remember that the parameter estimation (i. e., model selection) was performed on the validation data $\mathcal{D}_{\text{val}}$ while the final performance is evaluated on the test data $\mathcal{D}_{\text{test}}$. Generally, one would assume that an L^1 regularization increases the sparsification of potentials, although—in combination with the optimization of the regularization parameter on validation data with respect to the success rate—this is not guaranteed. In our experiments, L^1 regularization indeed removes the largest fraction of potentials on the hands, legs and spine sections dataset; only on the chests dataset, L^2 regularization removed more potentials than L^1. Remember that the optimization criterion from Eq. (7.1) is a non-linear non-convex problem, which requires to infer a new rival $\boldsymbol{s}_k^-$ in each iteration step. This—in combination with the grid optimization of θ related to the success rate—does not guarantee that L^1 regularization leads to a larger fraction of potentials being removed. The actual number of removed potentials is more of a beneficial side effect of optimizing the success rate.

Table 12.6: Comparison of different regularization methods w. r. t. the amount of removed potential functions (%) next to the achieved success rate (%) for the hands, legs, chests and spine sections datasets. Note that the reported results involve the optimization of the regularization parameter θ w. r. t. the success rate on the validation data $\mathcal{D}_{\text{val}}$; the selected values of θ are generally quite small ($\theta \leq 0.0004$ in all cases).

REGULARIZATION	POTENTIALS REMOVED (& SUCCESS RATE) / %			
	HANDS	LEGS	CHESTS	SPINE SECTIONS
None	54.8 (99.3)	35.3 (93.9)	69.4 (92.2)	**48.1** (85.2)
L^1	**75.2** (99.7)	**41.2** (94.0)	68.6 (91.3)	**48.1** (87.2)
L^2	59.0 (99.6)	35.3 (94.3)	**73.4** (90.4)	46.8 (85.3)

A further expected benefit of the sparsification of terms is the reduced CRF inference time. First, the number of potential functions is reduced and thus the amount of energy values to compute. Second, the topology is simplified, which ideally provides a search space (CRF states) that is better to traverse by the CRF inference. And indeed, the inference time has been reduced by on average $\sim 50\,\%$ on the first three datasets. The corresponding timings are listed in the first three rows of Table 12.7. Interestingly, the opposite effect has been observed for the spine sections datasets, where the inference time increased from 0.711 s to 2.269 s, although 48.1 % of the initial potential functions have been removed. This might be related to a much smoother energy land-

scape that negatively influences the A* heuristic and thus increases the effective search space of the algorithm.

Table 12.7: Listing of the inference times (potential computation plus CRF inference) before and after the optimization of the CRF achieved on the four datasets computed as averages over all test images.

DATASET	INFERENCE TIME / s		REL. CHANGE / %
	BEFORE OPT.	AFTER OPT.	
Hands	0.131	0.034	−74.0
Legs	0.027	0.017	−37.0
Chests	0.304	0.167	−45.1
Spine sections	0.711	2.269	+219.1

12.5 COMPARISON TO STATE OF THE ART

In order to put the previously illustrated results into perspective, they are compared to the state-of-the-art results reported in the literature that were achieved on the respective datasets.

12.5.1 *Hands*

We compare the results achieved on the hands dataset to the latest results published by Hahmann et al. [86]. In Table 12.8, the localization rates achieved by our method are compared to the ones reported in [86]. Generally, both methods are quite close and solve nearly all cases correctly. However, our method outperforms or is on par with the compared one on all key points except 1D and 5D.

Table 12.8: Comparison of previous results achieved by Hahmann et al. [86] on the hands dataset with the results achieved by our method in terms of localization rate of the individual key points and all key points for a localization tolerance of $R = {}^{6}/_{256} \cdot \text{image\_height}$.

METHOD	LOCALIZATION RATE / %												
	1D	2D	3D	5D	6D	7D	9D	10D	11D	13D	14D	15D	ALL
[86]	**99.5**	99.1	**99.3**	**99.7**	**99.3**	99.3	99.0	99.5	99.8	99.0	99.3	99.8	99.4
Ours	99.0	**99.3**	**99.3**	99.5	**99.3**	**99.8**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**99.7**

An average localization error of 6.6 px was reported in [86] for a mean image size of 1185×2006 px (a multi-scale pyramid was used to zoom from coarse to fine). Translating this error to a constant image height of 600 px (as used in our experiments) results in a localization error of

$\sim$ 1.92 px, which is worse than our achieved average localization error of 1.50 px.

A comparison to different methods operating on hand radiographs such as [184] or [151] is not possible due to, inter alia, incompatibilities of the evaluation measures and different datasets as well as target key points.

12.5.2 *Legs*

A direct comparison to the previous results of [168, 167] is difficult due to different evaluation setups (i. e., unknown training and test split versus 5-fold cross-validation) and a different objective (i. e., localization only versus detection and localization). However, if we only consider cases where an existing key point was detected (TP cases), we can quantify the localization performance of our approach and compare against the results achieved in [168]. Looking at the corresponding numbers listed in Table 12.9, we can see that our approach outperforms the previous results by a huge margin while solving a harder problem (i. e., detection and localization). Interestingly, both methods struggle with the hips and ankles in contrast to the knees. This might be related to the low contrast in the hip area and the general close proximity of the ankle key points to the image border.

Table 12.9: Comparison of previous results achieved by Ruppertshofen et al. [168] on the legs dataset with the results achieved by our method in terms of correctly localized key points averaged over the left and right side. Listed are the localization rates in percent (%) for a localization tolerance of $R = 10$ mm.

METHOD	LOCALIZATION RATE / %		
	HIP (HL + HR)	KNEE (KL + KR)	ANKLE (AL + AR)
Ruppertshofen et al. [168]	73.9	93.7	86.6
Ours	**94.1**	**99.1**	**91.6**

12.5.3 *Chests*

There are—to the best of our knowledge—no published results we can directly compare to. The only relevant work in this regard is the recently published extended abstract by Wessel et al. [204]. However, their use of a different non-disclosed dataset in combination with an incompatible evaluation scheme prevent a comparison, which has also been acknowledged in the public[1] reviews. Thus, the results presented

[1] The MIDL 2019 conference reviews of [204] are publicly available at https://openreview.net/forum?id=SJxuHzLjFV.

earlier stand on their own. However, we present and compare additional results achieved on the chests dataset in Chapter 13.

12.5.4 *Spine sections*

Recently, Payer et al. [151] found that 3 of the 60 test images were incorrectly labeled (spine chain shifts of up to 8 vertebrae) and informed the challenge organizers about it in order to update the dataset. While the previously stated results w. r. t. the spine sections dataset were computed against the corrected test set to give a proper impression of the method, we here first compare against other methods on the original test set (including mislabeled images) since all methods except for [151] were not aware of that incorrect labeling. Note that our results published in [134] were also computed against the mislabeled test images, which is why the previously stated values are better than the ones reported in [134].

Original Test Set

To compare the results to the current state of the art, we additionally calculated the identification rate as proposed in [77] (see Section 11.5). The results including the localization error are reported in Table 12.10. Note that the identification rate is agnostic to the detection task, i. e., it only considers predictions for the key points existing in the image. In contrast, the success rate accounts for all error types by normalizing to all landmarks (including the "missing" ones). As seen in Table 12.10, in terms of the localization rate, our approach (listed as "Ours") in its general form—i. e., not tuned to the vertebra localization task—does not achieve the results of other state-of-the-art algorithms. However, note that all listed approaches use more sophisticated multi-stage and shape-incorporating (e. g., LSTMs) deep neural networks that are arguably more complex in terms of runtime and memory demands in addition to explicitly requiring a recent GPU to operate in a reasonable timeframe, in contrast to our method. Only the approach in [78]—which is very close to the performance of our general approach—makes use of a non-deep-learning method in form of random forests (RFs) in combination with a shape model. Furthermore, all listed methods except for the very recent [151], which is a general shape-incorporating deep-learning-based approach, are specifically tuned towards the vertebra localization task.

Given the large number of parameters of the initial, fully connected CRF topology (910 potential weights plus 910 "missing" energies) which are estimated from just 145 training images, it is interesting to see how an informed, smaller initial topology compares. A good candidate is a chain structure connecting the neighboring nodes along

Table 12.10: Comparison of the previous results achieved on the original spine sections dataset (including three mislabeled images; see text) in contrast to the results achieved by our method in terms of the identification rate (in percent) and the localization error (average and standard deviation in mm). Note that these values are overly-pessimistic as they included mislabeled test images, which is—inter alia—evident in the high standard deviation.

METHOD	ID. RATE / %	LOCALIZATION ERROR / mm	
		MEAN	STD. DEV.
Glocker et al. [78]	74.4	13.2	17.8
Chen et al. [38]	84.0	8.8	13.0
Yang et al. [209]	85.0	8.7	**8.5**
Sekuboyina et al. [175]	86.1	7.4	9.3
Liao et al. [120]	88.3	6.5	8.6
Payer et al. [151]	90.9	**6.0**	16.1
Ours	73.4	11.9	19.5
Ours: chain	79.9	10.7	17.9
Ours: chain & conservative	**91.6**	6.2	16.2

the superoinferior axis, as done in [77], which only requires to estimate 126 + 126 parameters. Using this setup, a success rate of 86.8 % is achieved, which corresponds to 48.3 % fully correct images. In terms of identification rate (see "Ours: chain" in Table 12.10) this corresponds to 79.9 %, an improvement of 6.5 percent points. This indicates that the further incorporation of domain knowledge is beneficial.

A particular strength of our algorithm is its interpretability, e. g., in assessing the type of removed potential functions (see Section 12.4) and in interpreting the "missing" energies $\boldsymbol{B}$: The "missing" energies define an energy level beyond which the "missing" label $s_i = 0$ for a key point i is preferred over any of the n_i localization hypotheses $\mathcal{X}_i$. Thus, the CRF does not "trust" the n_i hypotheses generated by the localizer for that key point: no localization hypothesis $s_i > 0$ leads to a global key point configuration which has a lower energy than the configuration with the "missing" label $s_i = 0$ for key point i. This leads to a way of assessing the "confidence" of the CRF in the localization hypotheses in terms of the total energy of the key point configuration, Eq. (4.1). If the values of $\boldsymbol{B}$ are reduced, the "missing" label will be selected more often, especially for key points with a large contribution to the total energy from Eq. (4.1). This can be interpreted as if the CRF is less confident about the localization hypotheses $\mathcal{X}_i$ and selects the "missing" label instead, being more "conservative" in predicting valid key point positions. This may potentially increase the number of FN cases (and thus reduce the success rate, which accounts for all types of error), but it may increase the identification rate (which is agnostic to the detection problem) by potentially removing mis-localized predictions.

Conversely, if the values of $\boldsymbol{B}$ are increased, the "missing" label will be selected less often (more key point positions are predicted), reducing FN and TN cases while increasing FP and TP cases. Thus, by either scaling the automatically estimated values of $\boldsymbol{B}$ (or by manually specifying them), the operation point w. r. t. a receiver operating characteristic (ROC) curve can be modified, trading off FN and TN cases against FP and TP cases. This can be very helpful especially in a medical environment, preferring either a more conservative or a less conservative assignment of key point positions, depending on personal preference. To demonstrate this effect, we applied a scaling factor of $\gamma < 1$ to each individual value of $\boldsymbol{B}$ after CRF training (using the chain as initial CRF topology), thus favoring a more conservative setting with less predicted key points. The value of $\gamma = 0.775$ was chosen on the validation data as the lowest value for which a detection rate $^{(|\text{TN}|+|\text{TP}|)}/_{(K \cdot N)}$ of at least 75 % was achieved. By this scaling of $\boldsymbol{B}$, the success rate was reduced to 78.5 %, but the identification rate was significantly improved to 91.6 % (see row labeled "Ours: chain & conservative" in Table 12.10).

As mentioned earlier, the identification rate is agnostic to the detection task, hence the usage of additional metrics should be standard. However, only two of the reported methods provide (some) additional results next to the identification rate. Liao et al. [120] report a "classification accuracy" only for the second stage of their pipeline, a multi-task 3D CNN, the output of which is fed into a RNN which generates the final results. The reported classification accuracy of 52.4 % can be interpreted as a success rate with a weaker localization criterion and is still far behind our achieved 86.8 %. Arguably, the performance after the RNN should be better, however, the very low standard deviation of the localization error (Table 12.10) achieved on the original test set (including mislabeled images) indicates a potentially large number of FN cases and hence a bad detection rate despite the RNN. Sekuboyina et al. [175] provided numbers for precision ($^{|\text{L-TP}|}/_{(|\text{L-TP}|+|\text{M-TP}|+|\text{FP}|)}$) and recall ($^{|\text{L-TP}|}/_{(|\text{L-TP}|+|\text{M-TP}|+|\text{FN}|)}$) of 36.6 % and 87.9 %, respectively, for their method w. r. t. the identification criterion. Computing the same metrics for our method using the chain topology prior to the adjustments of $\boldsymbol{B}$ results in 76.9 % and 77.6 %, respectively, which indicates that [175] generates much more FP cases than our method.

Corrected Test Set

The authors of the most recently published results Payer et al. [151] found that three images of the test set were mislabeled (shifts along the spine by one or multiple vertebrae), which means that the results listed in Table 12.10—computed for the original dataset which included the three mislabeled test images—are probably over-pessimistic. See the last spine sections image in Fig. 12.2 illustrating an incorrectly annotated spine and the corresponding predictions, which are actually

fully correct w. r. t. the corrected annotation. Thus, they contacted the challenge organizers to correct the wrong annotations and published results computed on the original test set as well as the corrected test set. Note that all results listed in Table 12.10—including ours—were published prior to this discovery. Hence, we can only compare the results published in [151] to the reevaluated results of our approach, which are listed in Table 12.11. Interestingly, both approaches are very close for all three metrics on the original test set as well as the corrected test set, which might indicate that both methods struggle with the same set of (still incorrect) images and hence produce similar results. Furthermore, when we look at the standard deviation reported on the original test set in Table 12.10, we can see that Yang et al. [209], Sekuboyina et al. [175] and Liao et al. [120] reported comparably very low numbers, which might indicate that they either predicted incorrect but matching labels for the mislabeled images (shifts) or predicted many FN cases. The former seems rather unlikely as the mislabeled images are shifted by up to 8 vertebrae.

Table 12.11: Comparison of the results achieved by Payer et al. [151] on the corrected spine sections test set with the results achieved by our method in terms of the identification rate (in percent) and the localization error (average and standard deviation in mm). The previous results achieved on the original test set from Table 12.10 are shown in parentheses for comparison.

METHOD	ID. RATE / %	LOCALIZATION ERROR / mm	
		MEAN	STD. DEV.
Payer et al. [151]	96.0 (90.9)	2.9 (6.0)	4.4 (16.1)
Ours: chain & conservative	95.5 (91.6)	3.6 (6.2)	5.4 (16.2)

12.6 SUMMARY

We illustrated how the fairly simple combination of RTEs plus a CRF can be used to detect and localize key points in different datasets with different image modalities and target structures while making very few assumptions about the dataset at hand (only in terms of the target patch size), just by utilizing the necessary CRF optimization. While the performance prior to the optimization was far from ideal, we illustrated how the performance in terms of success rate and inference time is improved when automatically adapting the graph structure towards the target problem by removing unnecessary potential functions and reweighting them. Furthermore, as this is a benefit of model-based approaches, we showed how the resulting graph topologies might be interpreted w. r. t. the target dataset and align with our intuition. Furthermore, we showed how the results of the general and transferable

setup outperformed the previous results on three of the four dataset. This is not necessary to expect as we used few domain knowledge. By incorporating domain knowledge (in terms of an initial chain CRF topology instead of a fully connected graph and reducing the estimated "missing" energies), we further illustrated how the performance of our method can be improved to reach state-of-the-art results on the spine sections dataset.

13

OVERCOMING INSUFFICIENT LOCALIZATION HYPOTHESES

A major drawback of the method is the reduction of the search space to make inference feasible, i. e., considering only the n_i-best localization hypotheses (local maxima) for each key point rather than the whole image domain, as it potentially removes the correct solution from the search space. However, as outlined in Section 4.4, by adjusting the semantic meaning of the "missing" label, the insufficient localization hypotheses can be identified and resolved in a second CRF inference step.

13.1 EXPERIMENTAL SETUP

13.1.1 *Data*

This approach is evaluated on the chests dataset. Its low contrast in the area of the abdomen in combination with the hardly visible and hardly distinguishable posterior ribs makes it an ideal candidate for evaluation, as the local appearance models often fail in those areas and provide no correct key point prediction in the set of localization hypotheses.

13.1.2 *Parameters*

In contrast to the previous experiments, the modified U-Net as described in Section 5.2 is used to replace the RTEs as local appearance model. Remember that only one CNN is used to predict the heatmaps for all key points at the same time, which makes sense given the local ambiguity of the target key points and the potential benefit of shared features.

As the previous experiments suggested that a local connectivity pattern is a reasonable topology, we start the CRF optimization from a much sparser connectivity pattern, which is shown in Fig. 13.1. The same potential functions (i. e., binary potential functions evaluating distance, angle and vector) are used though, which results in three potential functions per edge and one potential function per node illustrated in Fig. 13.1.

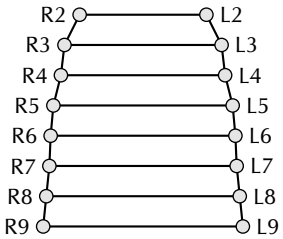


Figure 13.1: Illustration of the simplified graph topology used with the chests dataset. Circles correspond to graph nodes (with one unary potential function each) while edges indicate the usage of binary potential functions.

The training data (in each fold) is again split into three non-overlapping subsets $\mathcal{D}_{\text{pots}}$, $\mathcal{D}_{\text{graph}}$ and $\mathcal{D}_{\text{val}}$, containing however 50 %, 40 % and 10 % (instead of 30 %,

60 % and 10 %, respectively) of the K_{train} training images each. As before, the first subset $\mathcal{D}_{\text{pots}}$ is used to train potential functions, which includes the U-Net. Note that we assigned more training data to the estimation of the potentials due to the increased parameter size caused by the U-Net. The second subset $\mathcal{D}_{\text{graph}}$ is again used to optimize the CRF and estimate the energies for the "refine"—previously "missing"—label. The last subset $\mathcal{D}_{\text{val}}$ is used as validation set to select parameters such as regularization factor θ (an L^1 penalty is used) and learning rate η.

13.2 RESULTS

Applying the approach, 94.6 % of the key points were labeled correctly, corresponding to 83.0 % of the images where all 16 key points were correct. The localization rates computed for fully correct images and for key points at different steps of the approach are depicted in Fig. 13.2. First, we see that the CRF improves upon the plain U-Net results, especially in terms of the number of correct images. Second, we see that the U-Net provides few good alternative localization hypotheses, which is apparent in a bad upper bound (i. e., the theoretically best possible result caused by the reduction of the image domain to sets of localization hypotheses) of the CRF of just 59.7 % correct images and justifies the (third) refinement step. Third, we see that the additional CRF refinement step improves upon the plain CRF, where the percentage of correct cases increases dramatically from 57.3 % to 83.0 %. Fourth, the performance slightly decreases towards the lower ribs, which is probably caused by low contrast, higher variability and fewer meaningful surrounding structures in the abdomen (see for example the first image in Fig. 13.4b). A concrete example how the refinement overcomes this problem is illustrated in Fig. 13.3.

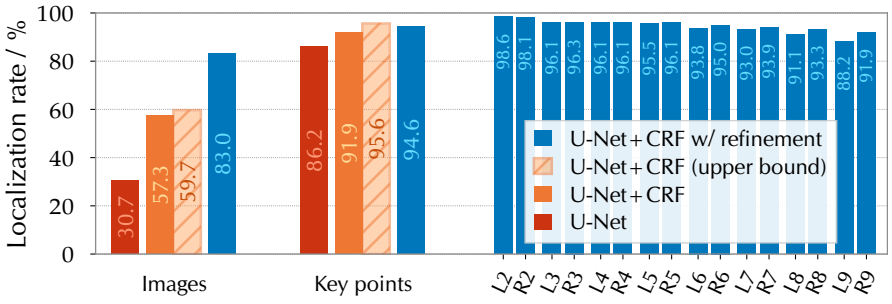
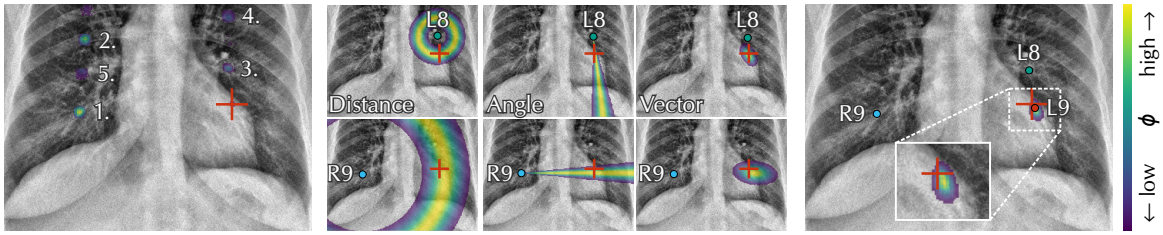


Figure 13.2: Illustration of the localization rates for fully correct images (i. e., all 16 key points localized correctly) and correctly localized key points for the three steps in percent. The upper bound indicates the theoretical maximal performance of the CRF, caused by the limitation of the state space to the set of localization hypotheses. Note that 100 % corresponds to 642 images, $642 \cdot 16 = 10\,272$ total key points and 642 instances for each individual key point (L2–R9).



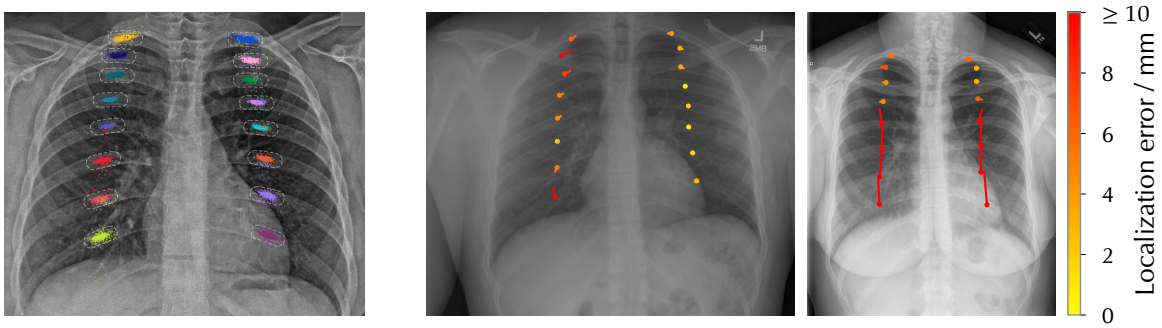
(a) Localization hypotheses (b) Binary potentials (c) Combined result

Figure 13.3: Illustration of the refinement process for L9 (annotated position indicated as ■ "+"). The "refine" label was chosen by the CRF inference for L9 because all n_i localization hypotheses—shown enumerated in (a); heatmap overlaid on cropped test image—yield large total energies $E(\boldsymbol{s})$. Utilizing the connected (b) binary potentials from L8 (■ circle) and R9 (■ circle), we are still able to (c) predict a correct position (■ circle) for L9 by evaluating over all image pixels instead of just the (here incorrect) localization hypotheses.

Localization errors in terms of Euclidean distance to the annotated position as well as Euclidean distance to the centerline are listed in Table 13.1. The resulting median values of 2.8 mm and 0.7 mm, respectively, are in line with the visualization of the prediction positions depicted in Fig. 13.4a. Note how the standard deviation increases when going from the second rib pair (L2+R2) towards the abdomen (L9+R9), which is in line with the key-point-specific localization rates from Fig. 13.2. The overall average runtime of the three steps comes down to 36 ms U-Net + 61 ms CRF + 73 ms refinement = 170 ms per image, respectively.

Table 13.1: Listing of the localization error as well as the Euclidean distance between the centerline and the corresponding predicted position (centerline distance) for each rib pair and all key points in mm. For both measures, the median, mean and standard deviation are listed.

KEY POINTS	LOCALIZATION ERROR / mm			CENTERLINE DISTANCE / mm		
	MEDIAN	MEAN	STD. DEV.	MEDIAN	MEAN	STD. DEV.
L2+R2	3.4	4.2	3.5	0.8	1.2	1.7
L3+R3	3.0	3.9	4.0	0.8	1.6	3.2
L4+R4	2.4	3.5	4.5	0.6	1.5	4.2
L5+R5	2.1	3.4	5.0	0.5	1.5	4.8
L6+R6	2.3	3.9	6.2	0.5	2.0	6.1
L7+R7	2.6	4.7	8.9	0.6	2.6	8.4
L8+R8	3.1	6.0	11.0	0.9	3.6	10.2
L9+R9	4.2	7.6	13.5	1.1	4.4	11.9
All	2.8	4.7	8.0	0.7	2.3	7.2



(a) Registered predictions (b) Typical errors: single mis-localizations and shifts

Figure 13.4: (a) Illustration of the final predicted positions in all 642 images by registering the images to a single (shown) image using the annotated positions (affine and b-spline) and warping the predicted positions into it; the warped positions are marked as color-coded dots. Furthermore, for each key point, the corresponding localization criterion has been visualized by the dashed elliptic line surrounding the allowed displacement. While the colored dots nicely visualize the predicted positions in a compact manner, they also introduce some errors caused by the registration. The image has been cropped and enhanced using adaptive histogram equalization. (b) Typical errors involve an incorrect localization in the abdomen as well as near the clavicle (first image) and chain errors caused by intermediate mistakes that cause a shift of key points along the ribs (second image). The circles indicate the annotated positions and (possibly too small to be seen) vectors are used to point towards the corresponding predicted positons. The error in mm of each prediction is color-coded.

13.3 COMPARISON

In contrast to the previous results from Section 12.2, the adjusted method improves upon all previous metrics as can be seen in Table 13.2. For instance, the amount of fully correct images has been increased from 72.0 % to 83.0 %.

Table 13.2: Comparison of two different methods, i. e., RTEs + CRF from Section 12.2 and the current method U-Net + CRF with refinement, in terms of localization rate (w. r. t. key points and fully correct images) and average Euclidean distance between the predicted positions and the corresponding annotated position as well as annotated centerline in mm.

METHOD	LOCALIZATION RATE / %		MEAN LOC. ERROR / mm	
	KEY POINTS	IMAGES	KEY POINTS	CENTERLINE
RTEs + CRF	92.2	72.0	5.6	2.6
U-Net + CRF w/ refine.	**94.6**	**83.0**	**4.7**	**2.3**

Applying the refinement when using a different local appearance model like the RTEs shows a similar increase in performance (see [131] for details[1]). However, it works especially well with overconfident

[1] The results published in [131]—although computed for the same dataset—were generated using a different experimental setup (e. g., no weight regularization) and are thus not directly comparable to the results listed here, which were published in [130].

methods such as the U-Net, which is illustrated in Table 13.3. The U-Net often generates a high-quality first localization hypothesis (labeled "first best" in Table 13.3) However, if this is not the case, the list of alternative localization hypotheses does often also not contain the correct location, so that the "cheating" localization rate (selecting the localization hypothesis from the set of localization hypotheses that is closest to the annotated position) increases only moderately. This strongly motivates to improve the set of localization hypotheses by the subsequent refinement step. The mentioned behavior is in contrast to the RTEs. They tend to provide a rather good candidate in the set of localization hypotheses associated with a lower probability (see the very good "cheating" results), but the "first best" candidate of the RTEs associated with a higher probability is often not the optimal one (so the "first best" results are rather low). Hence, it is more reasonable to use the refinement in combination with the U-Net rather than with the RTEs. Additionally, this indicates that the "confidence" of the local appearance model in the generated localization hypotheses is also very important and it is not sufficient to provide a correct localization hypotheses with a low probability. Note that the refinement also improves the performance of the RTEs, which has been evaluated in a detailed comparison in [131].

Table 13.3: Comparison of the RTEs and the U-Net in terms of localization rate (w. r. t. key points and fully correct images; in percent) computed for the first best localization hypothesis (first best) as well as the localization hypothesis—of the n_i localization hypotheses—that is closest to the annotated position (cheating).

METHOD	KEY POINT LOC. RATE / %		IMAGE LOC. RATE / %	
	FIRST BEST	CHEATING	FIRST BEST	CHEATING
RTEs	72.4	**98.8**	3.3	**86.9**
U-Net	**86.2**	95.6	**30.7**	59.7

13.4 SUMMARY

The localization and robust labeling of posterior ribs in thorax radiographs is a hard and unsolved problem [130]. A major problem in this setting is the superimposition of different structures, especially near the clavicle, as well as the very low contrast near the abdomen. Here, we set out to provide a robust and accurate approach by casting the rib labeling task to a localization task.

First, we illustrated that our method is agnostic to the local appearance model by exchanging the RTEs for a CNN in form of a slightly adapted U-Net architecture. As before, the CRF optimization greatly improved upon using only the localizer—U-Net in this case—in terms

of the localization performance. This is even more astonishing in case of the U-Net, as this network "sees" the whole image rather than just patches like in the case of the RTEs and is thus capable of integrating more context into its predictions.

Conversely, we have seen that the performance has an artificial upper bound caused by the reduction of the image domain to just a few localization hypotheses that are considered in the CRF inference. This is even more severe in case of the U-Net, as it tends to provide fewer good alternatives than the RTEs, which is related to the constrained field of view of the latter and the holistic view of the former. However, by using the "refine" label (a semantic adaptation of the "missing" label), we overcame the artificial upper bound by performing additional informed local inference steps on small subgraphs over the whole image domain. This greatly improved the localization rate (using the U-Net plus CRF) on image level from 57.3 % to 83.0 %, while the CRF performance is theoretically bounded to 59.7 % when not using the refinement. Note that this is a general approach and can be combined with arbitrary local appearance models and graph configurations and thus automatically benefits from further research in this area while still being applicable as a successive regularization step (see [131] for a more detailed evaluation).

14 HANDLING MANY KEY POINTS: A COMPARISON

Targeting the accurate localization of many (≥ 100) key points in 3D images in a reasonable time frame ($\leq 60\,\mathrm{s}$) is a hard problem. Only few approaches in the literature try to address it. Among them [216], which however requires a rigid structure, and [53], which provides very inaccurate predictions (high localization error). Thus, we evaluate our proposed method, which explicitly models spatial correlations in form of the CRF, on a highly variable structure, i. e., the spine, using 102 key points. In addition, we compare it to a dataset- and task-adapted state-of-the-art CNN that implicitly exploits the spatial correlation.

14.1 EXPERIMENTAL SETUP

14.1.1 *Data*

The full spines dataset is being used for evaluation. It provides 102 key points of locally ambiguous structures distributed over the spinal chain in CT images. The comparably small amount of ~ 125 training images per cross-validation instance additionally increases the severity of the task. For more details about the dataset see Section 10.5.

14.1.2 *RTEs+CRF: Parameters*

For each key point, we use an RTE local appearance model using a patch size of $\boldsymbol{A}_i = 67{\times}201{\times}201\,\mathrm{vx}$ (roughly $20{\times}20{\times}20\,\mathrm{cm}^3$) with $T_i = 32$ trees each and $V_i = 256$ features.

Since the initial experiments in Chapter 12 showed that a local connectivity pattern w. r. t. the anatomy is a preferred topology, a tree-structured connectivity pattern that naturally follows the kinematics of the spine is used (see Fig. 14.1). This not only simplifies the CRF parameterization quite a bit, but also improves CRF inference time since exact belief propagation might be used, which has linear time complexity. In addition to the unary RTE potential functions, the CRF is further parameterized by one vector potential function per connected key point pair (labeled "RTEs+CRF w/o latent scale" in Table 14.1). Instead of using any scaling or rotation invariant potential

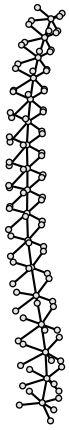


Figure 14.1: Illustration of the initial CRF topology using circles to indicate key points and connections between them to indicate binary potentials. The topology contains the spinal chain along the foramen to which the vertebra-specific key points are connected in a star-like pattern forming a tree.

functions, we evaluate this setup in contrast to the usage of a latent scaling variable $T^s \in \{0.80, 0.81, \dots, 1.20\}$ (labeled "RTEs+CRF w/ latent scale" in Table 14.1; see Section 6.4). Rotation is not considered as the images are taken in a fixed regime, while scaling is naturally introduced due to different patient heights.

The CRF optimization is carried out for 100 epochs using a margin of $m = 1$ and a learning rate of $\eta = 0.01$. In order to speed-up the training, no regularization (and thus no estimation of the regularization weight) is used. To select the rival $\boldsymbol{s}_k^-$, CRF inference in form of Gibbs sampling—an approximate MCMC-based method—is used. Note that the optimization is performed on 50 % of the training data, while the potentials (including the RTEs) were estimated on the other 50 %. This approach is later referred to as "RTEs+CRF". Note that no potential functions have been removed during the optimization.

14.1.3 *3D CPM: Parameters*

The previous setup is compared against the 3D CPM introduced in Chapter 8, which has a dedicated constellation module that tries to exploit key point co-occurrences. Over the course of 50 epochs, the network is trained using Adam with a learning rate of 1E−4 and the remaining parameters are used as suggested in [104]. Remember that 12 input patches are cut from each training image for processing. This resulted in a total training time of 2 to 3 days. In addition to comparing both approaches, we further evaluate the influence of the polynomial refinement (labeled "3D CPM w/ refinement" in Table 14.1; see Section 8.3) in contrast to the raw downsampled performance (labeled "3D CPM w/o refinement" in Table 14.1). Note that the downsampling is necessary to target datasets with such large amounts of key points. See Chapter 8 for a detailed explanation.

14.2 RESULTS

The results for both approaches are listed in Table 14.1. We see that the performance of both approaches is comparable with a slight advantage for the 3D CPM with a mean error of 4.32 mm after polynomial refinement (from an initial 5.86 mm prior to the refinement by taking just an additional 0.1 s), while requiring a 1.5 times longer test time than the RTEs+CRF (i. e., 45.4 s for the 3D CPM against 28.0 s for the RTEs+CRF). The latent scaling of the RTEs+CRF increases the key point localization rate from 90.7 % to 91.8 %, which is slightly better than the 3D CPM, and improves the mean error to 5.66 mm, which is still behind the 3D CPM though.

Looking at the localization error histogram depicted in Fig. 14.2, we can see that the better localization error (4.32 mm) of the 3D CPM is

Table 14.1: Results for the RTEs+CRF and the 3D CPM approach for localizing 102 key points on the full spines dataset averaged over the five folds in terms of localization rate (percentage of correct key points and of fully correct cases; correct if the localization error $\leq$ 10 mm), localization error in mm and average test time (per case) in seconds. For refinement and latent scale see text.

METHOD	LOC. RATE / %		LOC. ERROR / mm		TIME / s
	KEY POINTS	IMAGES	MEAN	STD. DEV.	
RTEs+CRF					
w/o latent scale	90.7	49.7	5.95	**9.19**	**24.8**
w/ latent scale	**91.8**	49.7	5.66	9.66	28.0
3D CPM					
w/o refinement	91.0	49.7	5.86	12.42	45.3
w/ refinement	91.1	**52.2**	**4.32**	12.66	45.4

caused by much more precise correct predictions despite the missing decoder path which is replaced by the polynomial refinement: the average localization error of correctly localized key points is 1.81 mm for the 3D CPM and 3.58 mm for the RTEs+CRF.

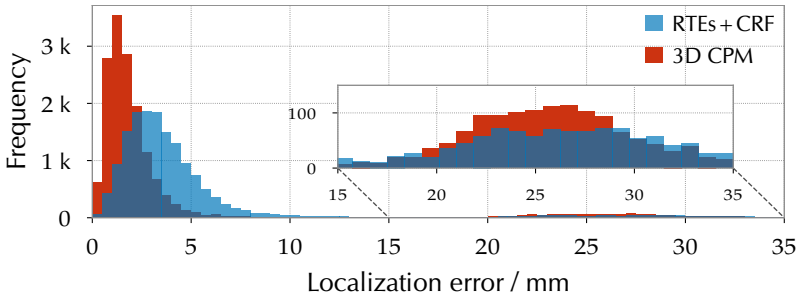


Figure 14.2: Histogram of the localization error in mm illustrating that the 3D CPM generates more precise key point predictions than the RTEs+CRF despite the strong downsampling that is countered by the polynomial refinement. Note the bump around 25 mm caused by vertebral confusions, which explains the higher standard deviation of the localization error of the 3D CPM than the RTEs+CRF listed in Table 14.1.

If we look how the mis-localizations are distributed over the spine by grouping the key points per associated vertebra (see Fig. 14.3), we can see that they are rather homogeneously distributed for the RTEs+CRF except for a slight increase towards the end (from L3 to L5) and more heterogeneously for the 3D CPM. From these observations we can conclude that the 3D CPM architecture is—as commonly assumed of deep networks—better in engineering features to generate accurate predictions, in contrast to the less powerful features used by the regression tree ensembles in the RTEs+CRF. However, the 3D CPM is not as good as the conditional random field in enforcing the global spatial correla-

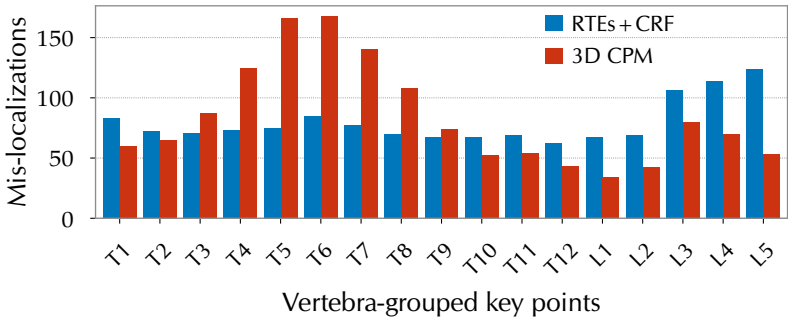


Figure 14.3: Illustration of the number of mis-localization per vertebra for the RTEs + CRF and the 3D CPM, which indicates that the CRF is better in enforcing the global spatial model than the 3D CPM.

tion between key points, which is additionally backed by the slightly lower standard deviation of the localization error for the RTEs+CRF (Table 14.1). This is in line with the qualitative results shown in Fig. 14.4 in form of typical mis-localizations. The 3D CPM rarely fails to predict a single key point but fails mostly in small sized groups of adjacent key points, whereas the CRF either fails for very small groups (i. e., no good local maximum in the RTE's heatmap) or shifts the whole CRF tree by one vertebra. Interestingly, both methods tend to fail for the same images, which indicates that both methods struggle—at least partly—with the same image characteristics.

14.3 FURTHER IMPROVEMENT

Both methods provide a design parameter to trade runtime with further improved performance. The RTEs+CRF allows to use more trees (> 32) to further improve the localization accuracy and quality of generated localization hypotheses, while the 3D CPM allows to use more stages to refine the predictions of the previous stages. Note that both do require more time to train as well as to test.

To increase the exploitation of the key point co-occurrence of the 3D CPM, we increased the number of stages to 3 (it is not possible to train more stages with the used hardware). As expected, this did improve upon the previous results (see Table 14.2), but the margin is comparably small given that the test time nearly doubled. To improve the sub-optimal local maxima found in the heatmaps in the RTEs+CRF, we increased the number of regression trees from 32 to $T_i = 48$ in the RTEs+CRF. As expected, this did improve the key point localization rate (0.8 %) while needing 7.9 s more for testing. It illustrates how the number of trees can be seen as adjustable parameter to trade an increased runtime for better localization accuracy. The decreased standard deviation of 6.7 mm supports the conclusion that the heatmap generation (RTEs) is the bottleneck of this approach in comparison to CNNs. How-

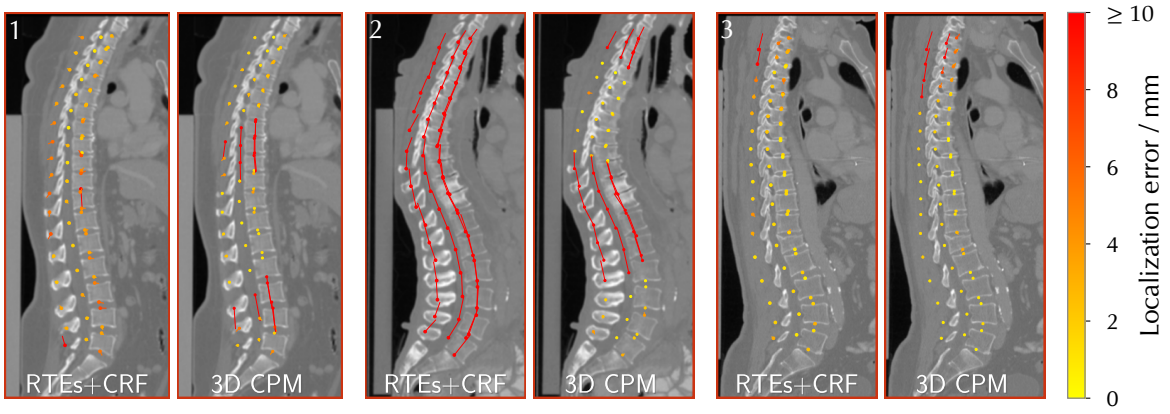


Figure 14.4: Illustration of typical errors comparing the RTEs + CRF and the 3D CPM side-by-side showing cropped sagittal slices of test images with circles indicating the annotated positions and (possibly too small to be seen) vectors pointing towards the corresponding predicted positons. The error in mm of each prediction is color-coded. Note that the processi transversus key points are not shown due to presentability. However, most errors are introduced by vertebra shifts, which apply equally to the processi transversus key points in case of failure. For more illustration of example predictions see the supplementary material of [132].

ever, note that the polynomial refinement used by the 3D CPM not only counters the downsampling but also the highly anisotropic voxels. The latter applies equally to the RTEs+CRF, but no polynomial refinement was used there. Thus, it is likely that applying it to the RTEs+CRF method would also reduce the localization error further.

Table 14.2: Results for the RTEs + CRF and the 3D CPM approach for localizing 102 key points on the full spines dataset averaged over the five folds in terms of localization rate (percentage of correct key points and of fully correct cases; correct if the localization error ≤ 10 mm), localization error in mm and average test time (per case) in seconds comparing the default setup against an improved setup of each method utilizing the respective design parameter.

METHOD	LOC. RATE / %		LOC. ERROR / mm		TIME / s
	KEY POINTS	IMAGES	MEAN	STD. DEV.	
RTEs + CRF w/ latent scale					
32 trees	91.8	49.7	5.66	9.66	**28.0**
48 trees	**92.6**	50.3	5.06	6.70	35.9
3D CPM w/ refinement					
2 stages	91.1	52.2	4.32	12.66	45.4
3 stages	91.4	**56.7**	**4.21**	12.51	88.9

14.4 SUMMARY

The localization of many repetitive key points in large CT images in a reasonable amount of time is a hard problem. Hence, there are few approaches in the literature that try to tackle that problem. Here, we

set out to provide a fair comparison between a state-of-the-art CNN-based approach and our CRF-based method. While the former tries to implicitly learn the constellation between key points, the latter tries to explicitly model them.

Both approaches have been tailored to some extent towards the task, i. e., the localization of 102 key points in spine CT images. The used CPM architecture has been adapted in terms of kernel-sizes and resulting receptive field to cover the best possible spine area while being trainable on a recent GPU with 12 GB of memory. As this resulted in heatmaps downsampled by a factor of 2×4×8, we suggested a refinement step based on fitting a second-order polynomial to derive continuous positions from the highly quantized downsampled and anisotropic voxels. For the second approach, a tree-based initial topology based on a local connectivity pattern was used to speed up the inference. Furthermore, a latent variable was used to model scaling of the spine in order to compensate for variable spine sizes as well as outliers.

We have seen that both approaches provide quite similar results, but also illustrated that they have different strengths and weaknesses. First, we saw that the RTEs+CRF is better in enforcing the global spatial structure as it models the spine structure in a holistic manner. In contrast, the 3D CPM is constrained in its view due to the necessary patching of the input image. Hence, in case of failure the 3D CPM tends to shift local key point groups by one vertebra while the RTEs+CRF tends to shift the whole spinal chain. With respect to the localization error, we saw that the 3D CPM produced more precise predictions, which is at least partly related to the polynomial refinement. In terms of runtime performance, the 3D CPM lacks behind by a factor of ~ 2, which is caused by the patching in combination with the necessary overlap and thus redundant processing of image data. For the RTEs+CRF, we also illustrated how it is possible to trade an increased runtime for a better localization performance, for example, by using 16 more trees (added to the initially 32 trees) the localization rate has been increased by ~ 1 percent point and the localization error has been reduced by ~ 0.5 mm, which further illustrates its practical usefulness.

15 APPLICATION: IVD SEGMENTATION

While key point detection is useful in itself in many applications, it is also beneficial for other follow-up tasks. Here, it is illustrated how a prior localization of small structures like the intervertebral discs (IVDs) may improve the segmentation accuracy of them as it allows to perform the segmentation in a highly reduced context. Additionally, scaling invariant ternary potential functions are compared to binary potential functions utilizing the latent scaling factor.

15.1 EXPERIMENTAL SETUP

15.1.1 *Data*

The lumbar spines dataset is being used for evaluation. It provides annotated segmentations of 7 IVDs for 16 images from which key point positions—i. e., the center of each IVD—were derived. The released lumbar spines dataset is used for a thorough evaluation, as we can perform a proper cross-validation (all 120 possible patient-split configurations at 2 images per patient). For evaluation on the 8 "non-disclosed lumbar spines", Docker[1] containers were prepared that contained our implementation and were used by the challenge organizers for fair evaluation. For more information about the dataset see Section 10.6.

15.1.2 *Parameters*

Two parameterizations of the CRF in terms of employed potential functions are compared. Again, RTEs are used as local appearance models (one per key point) with a patch size of roughly $7\times8.9\times8.9\,\mathrm{cm}^3$, 96 trees and 512 features (128 per channel). For the constellation, the first setup uses binary vector potential functions in combination with a latent scaling variable that are distributed along the spinal chain (tree structure; see left graph in Fig. 15.1). The second setup uses the implicitly scaling (and rotation) invariant ternary potential functions evaluating distances and angles (Section 6.2). Similarly to the previous setup, they are also distributed along the spinal

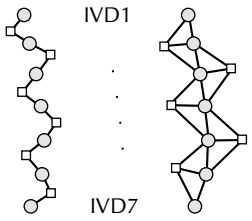


Figure 15.1: Illustration of the two initial topologies using binary vector potentials (left) and ternary distance ratio and relative angle potentials (right) on the lumbar spines dataset. Circles correspond to graph nodes (with one unary potential function each) while squares correspond to potential functions (potentially combined).

[1] The Docker platform uses operating-system-level virtualization to deliver software in packages called containers. For more information see https://www.docker.com.

chain, but do not form a tree though (see right graph in Fig. 15.1). The optimization is carried out as before, see [135] for more information.

15.1.3 *Segmentation*

The final goal is the segmentation of the IVD tissue. To do so, we use the localization as a first step in a multi-step approach, hence it is not of direct interest. Nevertheless, we evaluate the localization performance in detail later on in addition to the segmentation performance.

The segmentation is performed after the localization. First, small patches with constant size are cut around the localized key point positions from the volume and resampled to a standard orientation such that the IVDs are level inside the patches w. r. t. the axial plane. The used patch size of $7.4 \times 7.2 \times 3.1\,\mathrm{cm}^3$ was statistically estimated on the training data and is chosen such that the largest IVD is fully contained plus some safety margin to compensate for slight localization errors, while PCA was used to estimate the standard orientation of each IVD. Then, an off-the-shelf V-Net [139] is used to perform an IVD-agnostic (i. e., two-class) segmentation. The reasoning here is that the shapes and tissue patterns are very similar, and that it is thus reasonable to use an agnostic model. Finally, the resulting segmentations are projected back into the original-sized output segmentation and re-labeled according to the localizations. This whole pipeline is illustrated in Fig. 15.2.

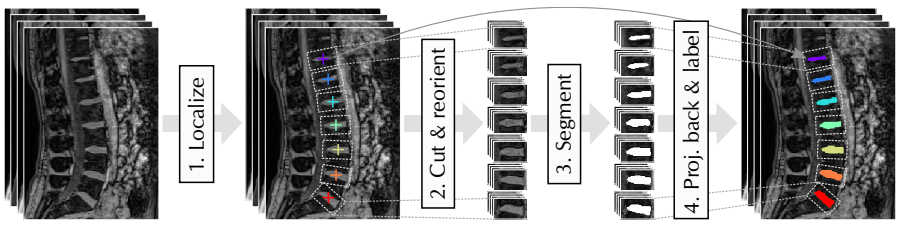


Figure 15.2: Illustration of the full segmentation pipeline starting from the initial key point localization to the final segmentation output.

In order to train the V-Net, the setup proposed by the authors Milletari et al. [139] with a mini-batch size of 7 (all IVDs at once) in combination with the generalized Dice loss [185] optimized by stochastic gradient descent in form of the Adam optimizer [104] was used. The training data was created by cutting patches around the derived IVD centroid positions. We also performed data augmentation in form of slight rotations ($\leq 10°$) and translations ($\leq 5\,\mathrm{mm}$) to create 24 additional augmented patches per original training patch, effectively increasing the training set size by a factor of 25. The optimization was carried out for 5 epochs (1750 iterations) with a learning rate of 1E−4.

Note that histogram matching [144] has been used prior to any processing to perform data normalization for each modality.

15.2 RESULTS ON RELEASED DATA

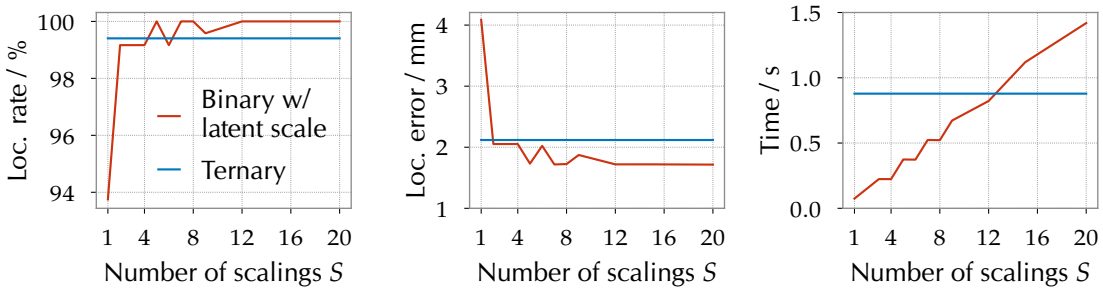
The following results were obtained on the 16 released images using the exhaustive cross-validation setup of 120 instances.

15.2.1 *Localization*

The localization performance w. r. t. the IVD centroids is compared between the two previously defined CRF setups. The average results over all 120 configurations are depicted in Fig. 15.3 for different numbers of scaling values, i. e., $T^s \in \{0.8 + i/s \cdot 0.4 \,|\, i \in [1\,..\,S]\} \cup \{1\}$ with S being the number of different scalings. Looking at the graphs we can see that scaling invariance has to be considered to achieve optimal performance. Furthermore, we see that the CRF setup that utilizes binary potentials with a latent scaling variable performs better than the setup using ternary potentials, evaluated over all metrics. It correctly localized the 7 IVDs in all test images in all 120 cross-validation instances with an average localization error of 1.72 mm for $S \geq 12$. In contrast, the CRF setup using ternary potentials failed in 3 of the 120 instances, mis-localizing 10 of the 1680 IVDs, resulting in a localization rate of 99.4 % and an average error of 2.12 mm. Looking at the runtime, we can see that the pairwise CRF potentials are faster to evaluate as long as $S \leq 12$ and provide better accuracy than the ternary CRF potentials if $S \geq 12$. In general, the increased amount of energy values to compute per potential ($n_i{}^3 \gg n_i{}^2$) plus the slower runtime of the A* inference algorithm make the ternary setup slower while not providing a better localization performance compared to pairwise CRF potentials as long as scaling invariance is accounted for in the latter setup. The number of scalings is set to $S = 100$ in the following.

15.2.2 *Segmentation*

The segmentation performance is assessed in terms of the Dice coefficient (similarity between annotated and predicated segmentation), the average surface distance (ASD) and center of mass distance between both segmentations (see [215] for more information) averaged over all test images over all 120 cross-validation instances. Again, the results of the two CRF setups are compared to analyze the influence of different CRF parameterizations on the final segmentation output, the results of which are listed in the first two rows in Table 15.1. Looking at the results, we can see that the better performance of the binary potentials in combination with a latent scaling variable manifested also in a better



(a) Localization rate (b) Localization error (c) Runtime

Figure 15.3: Comparison of the two different CRF setups— binary vector potentials with a latent scaling variable versus inherently scaling invariant ternary potentials—in terms of localization rate (in %; first graph), average localization error (in mm; second graph) and CRF computation plus inference time (in s; third graph) as a function of a different number S of values for the latent scaling variable s in the pairwise CRF potentials (the ternary potentials are independent of s).

final segmentation result with lower mean ASD and center of mass distance in combination much smaller standard deviation for all evaluation metrics. Also, we see that the small mean localization errors (1.72 mm and 2.12 mm for the binary and ternary CRF parameterizations, respectively) barely have an influence on the segmentation performance. This is further verified by looking at the achieved segmentation results when using the annotated key point positions as localization results to extract the patches used for segmentation (cheating experiment; last row in Table 15.1). We can see that all numbers are very much in the same interval, which indicates that the obtained localization performance is sufficient for an accurate segmentation. Further analysis revealed that by projecting the re-oriented, segmented IVDs back to the original image space, the Dice coefficient is slightly degraded, i. e., from 0.922 in re-oriented space to 0.903 in original space. This is likely caused by the small size of the volumes in relation to the surface of the IVDs and a voxel-based non-interpolating evaluation of the overlap, which results in an imprecise evaluation in the surface area.

Table 15.1: Evaluation of the full segmentation pipeline (see Fig. 15.2) comparing the two CRF parameterizations illustrated as mean ± standard deviations of the challenge evaluation metrics, averaged over all 120 cross-validation splits of the released lumbar spines data. The last row is a cheating experiment where the annotated (derived) positions were used instead of the predicted IVD centroid positions.

LOC. METHOD	DICE COEFF.	ASD / mm	CENTER DIST. / mm
CRF: binary w/ scaling	**0.904 ± 0.027**	**0.423 ± 0.085**	**0.590 ± 0.394**
CRF: ternary	0.902 ± 0.056	0.535 ± 2.238	0.715 ± 2.525
Annotated positions	0.903 ± 0.027	0.431 ± 0.087	0.640 ± 0.432

15.3 RESULTS ON NON-DISCLOSED DATA

The following results were obtained by the challenge organizers on the 8 non-disclosed images by using the prepared Docker containers.[2] For the Docker containers, we randomly selected 8 of the 120 cross-validation models, requiring balanced training (i. e., each of the 16 training images was used in exactly 7 cross-validation training runs), to form an ensemble of experts [153]. The final ensemble output is a voxel-wise majority vote on the (two-class) segmentation outputs of the 8 ensemble members. The idea is that the deficiencies of each model from the low number of training subjects cancel out. Again, we compare the two CRF parameterizations.

15.3.1 *Localization*

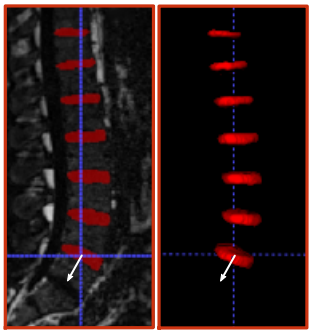


Figure 15.4: Test image with an incorrect shift of all segmentations towards the head by one IVD (left: sagittal plane; right: 3D rendering).

For the two cases 5 and 6 of the non-disclosed data, the entire IVD chain was shifted towards the head in both CRF configurations. An example[3] is shown in Fig. 15.4 with the white arrow indicating the correct start of the chain. Note that the IVD tissue was still accurately and precisely segmented. It is very surprising that the 8 localization models that produced fully correct cross-validation results majorly agreed that the chain should be shifted in both CRF parameterizations. Sadly, the reason for the assumed mis-localization cannot be determined, as the tested images are not disclosed and only the final segmentation output was reported. However, one might hypothesize that the spatial model overfit to the few training images, as the RTE for the lowest IVD near the sacrum produced very distinctive peaks without strong rivaling maxima for all visually inspected heatmaps of test images in the cross-validation setup, effectively forming an "anchor" for the chain. Although not evaluated on the non-disclosed data, reducing the amount of available localization hypotheses used for that last IVD to $n_i = 1$ still produced a localization rate of 100 %. If our assumption is correct, this should be an easy fix and prevent the shift on the non-disclosed data as well.

Furthermore, it illustrates another property of the method, namely the inclusion of expert knowledge. A common mode of operation in radiological practice is the inspection and potential correction of errors made by automatic algorithms. One might view fixing the set of localizing hypotheses to just one localization hypothesis, e. g., a manual correction. In this special case, the first localization hypothesis happened to be a correct one in all cases and thus might be considered expert knowledge, illustrating the idea.

[2] The submitted Docker containers and the corresponding results reported by the challenge organizers are publicly available at https://www.aomader.com/phd/.
[3] That one example image was kindly shared with us via e-mail by Guodong Zeng.

15.3.2 *Segmentation*

In terms of segmentation performance, the mis-localization reduced the overall Dice coefficient to 0.870 and 0.871 for the binary w/ latent variable and ternary CRF configuration, respectively. Note that overlapping IVDs are treated as correctly matched by the organizers. By selectively looking at the data we can get a better view on the actual segmentation performance, i. e., differentiating between localization and segmentation errors. We do so by excluding the 4 non-matching (i. e., not overlapping) IVDs of the total $8 \cdot 7 = 56$ IVDs. We compare these numbers to the best reported[4] numbers on the non-disclosed test data achieved by Georgiev and Asenov [72], computed over all 56 IVDs and the 56 minus 4 IVDs for fair comparison. Looking at the results listed in Table 15.2, we can see that when the mis-localized IVDs are excluded from the analysis, our full segmentation approach is on par (favorable) with the state of the art in terms of segmentation performance. Interestingly, the superiority of the binary setup in contrast to the ternary one suggested by the previous experiments is inverted (if only slightly) on the non-disclosed data, which might be caused by the ensemble setup and is statistically insignificant. Comparing the segmentation performance on the released data with the one on the non-disclosed data, we see an improvement of the Dice coefficient ($0.912 > 0.904$), which is likely caused by the ensemble setup.

Table 15.2: Comparison of our method to the currently best performing method on the non-disclosed lumbar spines data, again illustrating the mean $\pm$ the standard deviation of the challenge metrics. We ignore (indicated by a "-") values where an artificial very high penalty was introduced by the challenge organizers for mis-localized IVDs, which does not allow for a reasonable interpretation of those numbers.

METHOD	DICE COEFF.	ASD / mm	CENTER DIST. / mm
All 56 IVDs			
Georgiev et al. [72]	$\mathbf{0.911 \pm 0.024}$	$\mathbf{0.599 \pm 0.200}$	$\mathbf{0.756 \pm 0.404}$
Ours (binary w/ scaling)	0.870 ± 0.177	-	-
Ours (ternary)	0.871 ± 0.177	-	-
52 matching out of all 56 IVDs			
Georgiev et al. [72]	0.911 ± 0.025	0.602 ± 0.206	0.771 ± 0.408
Ours (binary w/ scaling)	$\mathbf{0.912 \pm 0.021}$	0.572 ± 0.153	0.741 ± 0.377
Ours (ternary)	$\mathbf{0.912 \pm 0.021}$	$\mathbf{0.570 \pm 0.153}$	$\mathbf{0.737 \pm 0.375}$

[4] The latest numbers are publicly available at https://ivdm3seg.weebly.com/results.html.

15.4 SUMMARY

As we have seen, the localization of small structures allows to segment these small structures in local context rather than the whole image, which theoretically provides an improved segmentation runtime (as only very small patches have to be segmented). In this experiment, we used our localization approach in combination with an off-the-shelf V-Net and simply followed the best practices to configure the segmentation network. In terms of segmentation performance, we illustrated that this approach is on par (favorable) to the best achieving method on the non-disclosed lumbar spines data. However, it has also been observed that the localization failed in two cases by shifting the whole chain towards the head by one IVD. To this end, it is not totally clear what caused this shift, but it might be related to overfitting on the training data in terms of the spatial constellation model (only 14 training images). Thus, a simple task-specific fix to overcome this problem was suggested.

However, the exhaustive cross-validation setup on the released lumbar spines data allowed for a more thorough evaluation. We have seen that incorporating scale by either a latent scaling variable or by the usage of inherently scaling invariant ternary potentials is beneficial in terms of localization performance, which further verifies our results regarding incorporating a scaling variable into the binary (pairwise) CRF potentials from Chapter 14. Regarding IVD segmentation on the released data, the binary vector potentials delivered a better result than the ternary ones though, which might also be related to the increased model complexity of the latter configuration (i. e., $5 \cdot 3$ ternary potential functions versus 6 binary potential functions). Note that on the released data, the CRF setup using binary potential functions in combination with the latent scaling variable localized all key points in all test images in our exhaustive cross-validation setup correctly.

16 CONCLUSIONS

In this thesis, a general and easy-to-transfer method for detecting and localizing spatially correlated key points in medical images based on a conditional random field (CRF) was suggested and thoroughly evaluated. The main problems of graph-based approaches, i. e., the dataset-dependent definition of potential functions in combination with the definition of a suitable topology, which hinder the general applicability, were circumvented by the proposed CRF optimization. Starting from a fully connected and fully loaded graph utilizing all potential functions from a pool of potential functions, the optimization removes unnecessary potential functions and weights them. Additionally, it adjusts the energies of a dedicated "missing" label in order to solve the detection problem in a principled way.

This has been evaluated on six different medical datasets with different imaging modalities (i. e., X-ray, CT and MRI), image dimensions (i. e., 2D and 3D) and different types and amount of target key points (ranging from 2 to 102). We illustrated how the general setup (i. e., very few domain assumptions) is already able to outperform previous state-of-the-art methods and how the method can be further geared towards specific datasets in order to reach the performance of highly sophisticated dataset-specific deep-learning-based approaches. Furthermore, we illustrated how the resulting models may be interpreted w. r. t. the dataset at hand and how it aligns with our intuition while illustrating the effect of the CRF optimization in terms of localization and runtime performance.

A compromise to reach decent CRF inference time is the reduction of the search space (the image domain) to sets of likely positions. As this might exclude the correct solution (correct position), it potentially limits the theoretical performance of the CRF. To overcome this problem, we proposed to change the semantic meaning of the "missing" label in order to identify insufficient sets of localization hypotheses, which allows to perform successive local inference runs on small subgraphs over the whole image domain rather than the set of localization hypotheses. We illustrated how this approach is able to overcome the theoretical upper bound (created by the reduced search space) of the localization performance on a previously unsolved problem, i. e., the localization and labeling of posterior ribs in chest radiographs. Note that this idea might apply to other CRF settings with large label sets besides key point detection and localization.

Furthermore, we compared different configurations of the CRF in terms of used local appearance models which define the unary CRF potentials and used pool of potential functions to illustrate the versatility of the approach. This included the usage of more sophisticated local appearance models in form of a CNN, scaling and rotation invariant higher order spatial potential functions and the exemplary modelling of scaling as a global transformation in terms of a latent variable. All of these were evaluated in terms of localization performance as well as runtime performance.

Additionally, we illustrated how other (follow-up) tasks such as segmentation may benefit from a prior localization of the target structures in case they are comparably small, as this allows to perform the segmentation on local context only. While we showed that such an approach generates a segmentation performance—evaluating only correctly localized structures—that is on par (favorably) with dedicated methods on the task of intervertebral disc (IVD) segmentation, it also means that the segmentation performance is tied to a correct localization of the structures to be segmented.

Finally, we would like to add that we were the first that provided a robust approach for the labeling of the posterior ribs in chest radiographs as well as localizing a very large amount of spinal key points in CT images in a reasonable amount of time. Both tasks are related to important medical problems and are thus of practical relevance.

16.1 FUTURE WORK

Although the obtained results are promising, the method might be further improved in some regards. In the following, we give some ideas for future research.

First, the CRF optimization is agnostic to the types of potential functions and hence it is interesting to incorporate further potential functions into the pool of potential functions. Promising types might use non-parametric approaches such as kernel density estimation to better model spatial statistics w. r. t. abnormal shapes (e. g., disease inflicted pathologies or dislocations) and incorrect assumptions about the underlying distributions.

Second, as we have shown, it is fairly trivial to use different kinds of local appearance models by exemplary using a U-Net CNN instead of multiple RTEs. Although the used CNN generated better results than the RTEs, it probably can be improved further. First of all, the receptive field size should be adaptable towards the target object rather than the whole image. This should not only generate more relevant localization hypotheses for the CRF to use in case of failure, but should also improve performance when having only few training images, which

is often the case in medical imaging. However, this is tightly coupled to the CNN architecture which is not easy to change without potentially destroying the performance. A potential solution for this problem has been proposed by Heinrich et al. [90], which might be a viable candidate for an easy-to-transfer CNN-based local appearance model.

Third, the joint optimization of potential function parameters is another direction for future research. However, note that this requires the potential functions to be differentiable in our optimization approach, which is not always the case though. Additionally, this might become intractable quickly with many key points, high resolution 3D images and the usage of sophisticated differentiable models such as CNNs. An initial experiment in this direction is presented in Appendix A.

BIBLIOGRAPHY

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. "TensorFlow: A System for Large-Scale Machine Learning." In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. USENIX Association, 2016, pp. 265–283.

[2] A. Alansary, O. Oktay, Y. Li, L. Le Folgoc, B. Hou, G. Vaillant, K. Kamnitsas, A. Vlontzos, B. Glocker, B. Kainz, et al. "Evaluating reinforcement learning agents for anatomical landmark detection." In: *Medical Image Analysis* 53 (Apr. 2019), pp. 156–164.

[3] M. Alber, S. Lapuschkin, P. Seegerer, M. Hägele, K. T. Schütt, G. Montavon, W. Samek, K.-R. Müller, S. Dähne, and P.-J. Kindermans. "iNNvestigate neural networks." In: *Journal of Machine Learning Research* 20.93 (2019), pp. 1–8.

[4] K. Amplianitis, R. Hänsch, and R. Reulke. "Human Recognition in RGBD Combining Object Detectors and Conditional Random Fields." In: *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. Scitepress, 2016, pp. 655–663.

[5] B. Andres, T. Beier, and J. H. Kappes. *OpenGM: A C++ Library for Discrete Graphical Models*. 2012. arXiv: `1206.0111 [cs.AI]`.

[6] Y. Artan, M. A. Haider, D. L. Langer, T. H. van der Kwast, A. J. Evans, Y. Yang, M. N. Wernick, J. Trachtenberg, and I. S. Yetik. "Prostate Cancer Localization With Multispectral MRI Using Cost-Sensitive Support Vector Machines and Conditional Random Fields." In: *IEEE Transactions on Image Processing* 19.9 (Sept. 2010), pp. 2444–2455.

[7] K. Babalola and T. Cootes. "Using parts and geometry models to initialise Active Appearance Models for automated segmentation of 3D medical images." In: *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. IEEE, 2010, pp. 1069–1072.

[8] D. Ballard. "Generalizing the Hough Transform to Detect Arbitrary Shapes." In: *Readings in Computer Vision*, ed. by M. A. Fischler et al. Morgan Kaufmann, 1987, pp. 714–725.

[9] T. Baltrusaitis, P. Robinson, and L.-P. Morency. "Constrained Local Neural Fields for Robust Facial Landmark Detection in the Wild." In: *2013 IEEE International Conference on Computer Vision Workshops*. IEEE, 2013, pp. 354–361.

[10] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra. "Clustering on the Unit Hypersphere using von Mises-Fisher Distributions." In: *Journal of Machine Learning Research* 6 (Sept. 2005), pp. 1345–1382.

[11] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich. "Diverse M-Best Solutions in Markov Random Fields." In: *Computer Vision – ECCV 2012*, ed. by A. Fitzgibbon et al. Springer, 2012, pp. 1–16.

[12] H. Bay, T. Tuytelaars, and L. Van Gool. "SURF: Speeded up robust features." In: *Computer Vision – ECCV 2006*, ed. by A. Leonardis et al. Lecture Notes in Computer Science 3951. Springer, 2006, pp. 404–417.

[13] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind. "Automatic Differentiation in Machine Learning: A Survey." In: *Journal of Machine Learning Research* 18.153 (2018), pp. 5595–5637.

[14] A. Beinglass and H. J. Wolfson. "Articulated Object Recognition, or: How to Generalize the Generalized Hough Transform." In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1991, pp. 461–466.

[15] J. von Berg, C. Levrier, H. Carolus, S. Young, A. Saalbach, P. Laurent, and R. Florent. "Decomposing the bony thorax in X-ray images." In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2016, pp. 1068–1071.

[16] M. Bergtholdt, J. Kappes, S. Schmidt, and C. Schnörr. "A study of parts-based object class detection using complete graphs." In: *International Journal of Computer Vision* 87.1 (Jan. 2009), pp. 87–93.

[17] M. Bergtholdt, J. Kappes, and C. Schnörr. "Learning of Graphical Models and Efficient Inference for Object Class Recognition." In: *Pattern Recognition*, ed. by K. Franke et al. Lecture Notes in Computer Science 4174. Springer, 2006, pp. 273–283.

[18] J. G. Betts, K. A. Young, J. A. Wise, E. Johnson, B. Poe, D. H. Kruse, O. Korol, J. E. Johnson, M. Womble, and P. DeSaix. *Anatomy and Physiology*. OpenStax, 2013.

[19] J. Beutel, H. L. Kundel, and R. L. van Metter, eds. *Handbook of Medical Imaging. Physics and Psychophysics*. Vol. 1. SPIE Press, 2000.

[20] B. Bier, M. Unberath, J.-N. Zaech, J. Fotouhi, M. Armand, G. Osgood, N. Navab, and A. Maier. "X-ray-transform Invariant Anatomical Landmark Detection for Pelvic Trauma Surgery." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11073. Springer, 2018, pp. 55–63.

[21] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[22] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. "Visual object tracking using adaptive correlation filters." In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2544–2550.

[23] T. Bouwmans, C. Silva, C. Marghes, M. S. Zitouni, H. Bhaskar, and C. Frelicot. "On the role and the importance of features for background modeling and foreground detection." In: *Computer Science Review* 28 (2018), pp. 26–91.

[24] L. Breiman. "Bagging predictors." In: *Machine Learning* 24.2 (Aug. 1996), pp. 123–140.

[25] L. Breiman. "Random Forests." In: *Machine Learning* 45.1 (Oct. 2001), pp. 5–32.

[26] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Taylor & Francis, 1984.

[27] F. Bromberg, D. Margaritis, and V. Honavar. "Efficient Markov network structure discovery using independence tests." In: *Journal of Artificial Intelligence Research* 35 (July 2009), pp. 449–484.

[28] M. Brunk, H. Ruppertshofen, S. Schmidt, P. Beyerlein, and H. Schramm. "Bone Age Classification Using the Discriminative Generalized Hough Transform." In: *Bildverarbeitung für die Medizin 2011*, ed. by H. Handels et al. Informatik aktuell. Springer, 2011, pp. 284–288.

[29] C. Buerger, J. von Berg, A. Franz, T. Klinder, C. Lorenz, and M. Lenga. "Combining deep learning and model-based segmentation for labeled spine CT segmentation." In: *Medical Imaging 2020: Image Processing*, ed. by I. Išgum et al. Vol. 11313. SPIE, 2020, pp. 307–314.

[30] C. Burgmer. *Diagram of an artificial neuron*. 2005. Available at: `https://commons.wikimedia.org/wiki/File:ArtificialNeuronModel.png` (visited on 05/22/2020).

[31] M. C. Burl, M. Weber, and P. Perona. "A probabilistic approach to object recognition using local photometry and global geometry." In: *Computer Vision — ECCV'98*, ed. by H. Burkhardt et al. Lecture Notes in Computer Science 1407. Springer, 1998, pp. 628–641.

[32] N. Cai, H. Chen, Y. Li, Y. Peng, J. Li, and X. Li. "Reducing non-realistic deformations in registration using precise and reliable landmark correspondences." In: *Computers in Biology and Medicine* 115 (Dec. 2019).

[33] J. C. Caicedo and S. Lazebnik. "Active Object Localization with Deep Reinforcement Learning." In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 2488–2496.

[34] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. "BRIEF: Binary Robust Independent Elementary Features." In: *Computer Vision – ECCV 2010*, ed. by K. Daniilidis et al. Lecture Notes in Computer Science 6314. Springer, 2010, pp. 778–792.

[35] S. Candemir, S. Jaeger, S. Antani, U. Bagci, L. R. Folio, Z. Xu, and G. Thoma. "Atlas-based rib-bone detection in chest X-rays." In: *Computerized Medical Imaging and Graphics* 51 (July 2016), pp. 32–39.

[36] G. Carneiro, B. Georgescu, S. Good, and D. Comaniciu. "Detection and Measurement of Fetal Anatomies from Ultrasound Images using a Constrained Probabilistic Boosting Tree." In: *IEEE Transactions on Medical Imaging* 27.9 (July 2008), pp. 1342–1355.

[37] C. Chen, D. L. Belavý, W. Yu, C. Chu, G. Armbrecht, M. Bansmann, D. Felsenberg, and G. Zheng. "Localization and Segmentation of 3D Intervertebral Discs in MR Images by Data Driven Estimation." In: *IEEE Transactions on Medical Imaging* 34.8 (Aug. 2015), pp. 1719–1729.

[38] H. Chen, C. Shen, J. Qin, D. Ni, L. Shi, J. C. Y. Cheng, and P.-A. Heng. "Automatic Localization and Identification of Vertebrae in Spine CT via a Joint Learning Model with Deep Neural Networks." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, ed. by N. Navab et al. Lecture Notes in Computer Science 9349. Springer, 2015, pp. 515–522.

[39] K. Chen, C. Peng, Y. Li, D. Cheng, and S. Wei. "Accurate Automated Keypoint Detections for Spinal Curvature Estimation." In: *Computational Methods and Clinical Applications for Spine Imaging*, ed. by Y. Cai et al. Lecture Notes in Computer Science 11963. Springer, 2020, pp. 63–68.

[40] R. Chen, Y. Ma, N. Chen, D. Lee, and W. Wang. "Cephalometric Landmark Detection by Attentive Feature Pyramid Fusion and Regression-Voting." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11766. Springer, 2019, pp. 873–881.

[41] X. Chen and A. L. Yuille. "Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations." In: *Advances in Neural Information Processing Systems 27*, ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 1736–1744.

[42] Y. Chen, C. Shen, H. Chen, X.-S. Wei, L. Liu, and J. Yang. "Adversarial Learning of Structure-Aware Fully Convolutional Networks for Landmark Localization." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Feb. 2019).

[43] Y. Chen, C. Shen, X.-S. Wei, L. Liu, and J. Yang. "Adversarial PoseNet: A Structure-Aware Convolutional Network for Human Pose Estimation." In: *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 1221–1230.

[44] V. Cheplygina, M. de Bruijne, and J. P. W. Pluim. "Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis." In: *Medical Image Analysis* 54 (May 2019), pp. 280–296.

[45] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, ed. by S. Ourselin et al. Lecture Notes in Computer Science 9901. Springer, 2016, pp. 424–432.

[46] T. F. Cootes, G. J. Edwards, and C. J. Taylor. "Active Appearance Models." In: *Computer Vision — ECCV'98*, ed. by H. Burkhardt et al. Lecture Notes in Computer Science 1407. Springer, 1998, pp. 484–498.

[47] T. F. Cootes, G. J. Edwards, and C. J. Taylor. "Active Appearance Models." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.6 (June 2001), pp. 681–685.

[48] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. "The Use of Active Shape Models For Locating Structures in Medical Images." In: *Image and Vision Computing* 12.6 (July 1994), pp. 355–365.

[49] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. "Active Shape Models—Their Training and Application." In: *Computer Vision and Image Understanding* 61.1 (Jan. 1995), pp. 38–59.

[50] T. F. Cootes and C. J. Taylor. *Statistical Models of Appearance for Computer Vision*. Tech. rep. University of Manchester, 2004.

[51] A. Criminisi, D. Robertson, E. Konukoglu, J. Shotton, S. Pathak, S. White, and K. Siddiqui. "Regression forests for efficient anatomy detection and localization in computed tomography scans." In: *Medical Image Analysis* 17.8 (Dec. 2013), pp. 1293–1303.

[52] D. Cristinacce, T. F. Cootes, and I. M. Scott. "A Multi-Stage Approach to Facial Feature Detection." In: *Proceedings of the British Machine Vision Conference*. 10.5244/C.18.30. BMVA Press, 2004, pp. 277–286.

[53] M. A. Dabbah, S. Murphy, H. Pello, R. Courbon, E. Beveridge, S. Wiseman, D. Wyeth, and I. Poole. "Detection and location of 127 anatomical landmarks in diverse CT datasets." In: *Medical Imaging 2014: Image Processing*, ed. by S. Ourselin et al. Vol. 9034. SPIE, 2014, pp. 284–294.

[54] D. Damopoulos, B. Glocker, and G. Zheng. "Automatic Localization of the Lumbar Vertebral Landmarks in CT Images with Context Features." In: *Computational Methods and Clinical Applications in Musculoskeletal Imaging*, ed. by B. Glocker et al. Lecture Notes in Computer Science 10734. Springer, 2018, pp. 59–71.

[55] D. Demner-Fushman, S. Antani, M. Simpson, and G. R. Thoma. "Design and Development of a Multimodal Biomedical Information Retrieval System." In: *Journal of Computing Science and Engineering* 6.2 (June 2012), pp. 168–177.

[56] D. Demner-Fushman, M. D. Kohli, M. B. Rosenman, S. E. Shooshan, L. Rodriguez, S. Antani, G. R. Thoma, and C. J. McDonald. "Preparing a collection of radiology examinations for distribution and retrieval." In: *Journal of the American Medical Informatics Association* 23.2 (Mar. 2016), pp. 304–310.

[57] W. T. Dixon. "Simple proton spectroscopic imaging." In: *Radiology* 153.1 (Oct. 1984), pp. 189–194.

[58] R. Donner, G. Langs, B. Mičušík, and H. Bischof. "Generalized sparse MRF appearance models." In: *Image and Vision Computing* 28.6 (June 2010), pp. 1031–1038.

[59] R. Donner, B. H. Menze, H. Bischof, and G. Langs. "Global localization of 3D anatomical structures by pre-filtered Hough Forests and discrete optimization." In: *Medical Image Analysis* 17.8 (Mar. 2013), pp. 1304–1314.

[60] R. Donner, B. Mičušík, G. Langs, and H. Bischof. "Sparse MRF Appearance Models for Fast Anatomical Structure Localisation." In: *Proceedings of the British Machine Vision Conference*. BMVA Press, 2007, pp. 109.1–109.10.

[61] R. O. Duda and P. E. Hart. "Use of the Hough Transformation to Detect Lines and Curves in Pictures." In: *Communications of the ACM* 15.1 (Jan. 1972), pp. 11–15.

[62] V. Dumoulin and F. Visin. *A guide to convolution arithmetic for deep learning*. 2018. arXiv: `1603.07285 [stat.ML]`.

[63] O. Ecabert, J. Peters, H. Schramm, C. Lorenz, J. von Berg, M. J. Walker, M. Vembar, M. E. Olszewski, K. Subramanyan, G. Lavi, et al. "Automatic model-based segmentation of the heart in CT images." In: *IEEE Transactions on Medical Imaging* 27.9 (Sept. 2008), pp. 1189–1201.

[64] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. "Object Detection with Discriminatively Trained Part-Based Models." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.9 (Sept. 2010), pp. 1627–1645.

[65] P. F. Felzenszwalb and D. P. Huttenlocher. "Pictorial Structures for Object Recognition." In: *International Journal of Computer Vision* 61.1 (Jan. 2005), pp. 55–79.

[66] M. Fenchel, S. Thesen, and A. Schilling. "Automatic Labeling of Anatomical Structures in MR FastView Images Using a Statistical Atlas." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008*, ed. by D. Metaxas et al. Lecture Notes in Computer Science 5241. Springer, 2008, pp. 576–584.

[67] X. Feng, J. Yang, A. F. Laine, and E. D. Angelini. "Discriminative Localization in CNNs for Weakly-Supervised Segmentation of Pulmonary Nodules." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10435. Springer, 2017, pp. 568–576.

[68] B. Fischer, A. Brosig, T. M. Deserno, B. Ott, and R. W. Günther. "Structural scene analysis and content-based image retrieval applied to bone age assessment." In: *Medical Imaging 2009: Computer-Aided Diagnosis*, ed. by N. Karssemeijer et al. Vol. 7260. SPIE, 2009, pp. 39–49.

[69] M. A. Fischler and R. A. Elschlager. "The Representation and Matching of Pictorial Structures." In: *IEEE Transactions on Computers* C-22.1 (Jan. 1973), pp. 67–92.

[70] Y. Freund and R. E. Schapire. "A desicion-theoretic generalization of on-line learning and an application to boosting." In: *Computational Learning Theory*, ed. by P. Vitányi. Lecture Notes in Artificial Intelligence 904. Springer, 1995, pp. 23–37.

[71] J. Gall and V. Lempitsky. "Class-Specific Hough Forests for Object Detection." In: *Decision Forests for Computer Vision and Medical Image Analysis*, ed. by A. Criminisi et al. Springer, 2013, pp. 143–157.

[72] N. Georgiev and A. Asenov. "Automatic Segmentation of Lumbar Spine MRI Using Ensemble of 2D Algorithms." In: *Computational Methods and Clinical Applications for Spine Imaging*, ed. by G. Zheng et al. Lecture Notes in Computer Science 11397. Springer, 2019, pp. 154–162.

[73] P. Geurts, D. Ernst, and L. Wehenkel. "Extremely randomized trees." In: *Machine Learning* 63.1 (Apr. 2006), pp. 3–42.

[74] F. C. Ghesu, E. Krubasik, B. Georgescu, V. Singh, Y. Zheng, J. Hornegger, and D. Comaniciu. "Marginal Space Deep Learning: Efficient Architecture for Volumetric Image Parsing." In: *IEEE Transactions on Medical Imaging* 35.5 (May 2016), pp. 1217–1228.

[75] F. C. Ghesu, B. Georgescu, S. Grbic, A. Maier, J. Hornegger, and D. Comaniciu. "Towards intelligent robust detection of anatomical structures in incomplete volumetric data." In: *Medical Image Analysis* 48 (June 2018), pp. 203–213.

[76] F. C. Ghesu, B. Georgescu, Y. Zheng, S. Grbic, A. Maier, J. Hornegger, and D. Comaniciu. "Multi-Scale Deep Reinforcement Learning for Real-Time 3D-Landmark Detection in CT Scans." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.1 (Jan. 2019), pp. 176–189.

[77] B. Glocker, J. Feulner, A. Criminisi, D. R. Haynor, and E. Konukoglu. "Automatic Localization and Identification of Vertebrae in Arbitrary Field-of-View CT Scans." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*, ed. by N. Ayache et al. Lecture Notes in Computer Science 7512. Springer, 2012, pp. 590–598.

[78] B. Glocker, D. Zikic, E. Konukoglu, D. R. Haynor, and A. Criminisi. "Vertebrae Localization in Pathological Spine CT via Dense Classification from Sparse Annotations." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, ed. by K. Mori et al. Lecture Notes in Computer Science 8150. Springer, 2013, pp. 262–270.

[79] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[80] A. Gooßen, E. Hermann, G. M. Weber, T. Gernoth, T. Pralow, and R.-R. Grigat. "Model-based segmentation of pediatric and adult joints for orthopedic measurements in digital radiographs of the lower limbs." In: *Computer Science - Research and Development* 26.1 (Feb. 2011), pp. 107–116.

[81] A. Gooßen, M. Schlüter, T. Pralow, and R.-R. Grigat. "A Stitching Algorithm for Automatic Registration of Digital Radiographs." In: *Image Analysis and Recognition*, ed. by A. Campilho et al. Lecture Notes in Computer Science 5112. Springer, 2008, pp. 854–862.

[82] G. R. Grimmett. "A Theorem about Random Fields." In: *Bulletin of the London Mathematical Society* 5.1 (Mar. 1973), pp. 81–84.

[83] G. Guglielmi, S. Muscarella, and A. Bazzocchi. "Integrated Imaging Approach to Osteoporosis: State-of-the-Art Review and Update." In: *RadioGraphics* 31.5 (Sept. 2011), pp. 1343–1364.

[84] K. Gurney. *An Introduction to Neural Networks*. Routledge, 1997.

[85] F. Hahmann, I. Berger, H. Ruppertshofen, T. M. Deserno, and H. Schramm. "Bone Age Assessment Using the Classifying Generalized Hough Transform." In: *Pattern Recognition*, ed. by J. Weickert et al. Lecture Notes in Computer Science 8142. Springer, 2013, pp. 313–322.

[86] F. Hahmann, G. Böer, E. Gabriel, T. M. Deserno, C. Meyer, and H. Schramm. "Classification of voting patterns to improve the generalized Hough transform for epiphyses localization." In: *Medical Imaging 2016: Computer-Aided Diagnosis*, ed. by G. D. Tourassi et al. Vol. 9785. SPIE, 2016, pp. 47–57.

[87] J. M. Hammersley and P. Clifford. "Markov fields on finite graphs and lattices." Unpublished Berkeley preprint. 1971.

[88] H. Hebelka, L. Torén, K. Lagerstrand, and H. Brisby. "Axial loading during MRI reveals deviant characteristics within posterior IVD regions between low back pain patients and controls." In: *European Spine Journal* 27.11 (Nov. 2018), pp. 2840–2846.

[89] M. P. Heinrich and O. Oktay. "Accurate Intervertebral Disc Localisation and Segmentation in MRI Using Vantage Point Hough Forests and Multi-atlas Fusion." In: *Computational Methods and Clinical Applications for Spine Imaging*, ed. by J. Yao et al. Lecture Notes in Computer Science 10182. Springer, 2016, pp. 77–84.

[90] M. P. Heinrich, O. Oktay, and N. Bouteldja. "OBELISK-Net: Fewer layers to solve 3D multi-organ segmentation with sparse deformable convolutions." In: *Medical Image Analysis* 54 (May 2019), pp. 1–9.

[91] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. "High-Speed Tracking with Kernelized Correlation Filters." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.3 (Mar. 2015), pp. 583–596.

[92] A. Hill, T. F. Cootes, C. J. Taylor, and K. Lindley. "Medical image interpretation: A generic approach using deformable templates." In: *Medical Informatics* 19.1 (1994), pp. 47–59.

[93] G. E. Hinton. "Training Products of Experts by Minimizing Contrastive Divergence." In: *Neural computation* 14.8 (Aug. 2002), pp. 1771–1800.

[94] P. V. C. Hough. "Method and means for recognizing complex patterns." U.S. pat. 3 069 654 (US). Dec. 18, 1962.

[95] B. Howe, A. Gururajan, H. Sari-Sarraf, and L. R. Long. "Hierarchical segmentation of cervical and lumbar vertebrae using a customized generalized Hough transform and extensions to active appearance models." In: *6th IEEE Southwest Symposium on Image Analysis and Interpretation, 2004.* IEEE, 2004, pp. 182–186.

[96] B. Hurley, B. O'Sullivan, D. Allouche, G. Katsirelos, T. Schiex, M. Zytnicki, and S. de Givry. "Multi-language evaluation of exact solvers in graphical model discrete optimization." In: *Constraints* 21.3 (July 2016), pp. 413–434.

[97] M. A. Hussain, A. Amir-Khalili, G. Hamarneh, and R. Abugharbieh. "Segmentation-Free Kidney Localization and Volume Estimation Using Aggregated Orthogonal Decision CNNs." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10435. Springer, 2017, pp. 612–620.

[98] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P. F. Jaeger, S. Kohl, J. Wasserthal, G. Koehler, T. Norajitra, S. Wirkert, et al. *nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation*. 2018. arXiv: 1809.10486 [cs.CV].

[99] Z. Jie, X. Liang, J. Feng, X. Jin, W. Lu, and S. Yan. "Tree-Structured Reinforcement Learning for Sequential Object Localization." In: *Advances in Neural Information Processing Systems 29*, ed. by D. D. Lee et al. Curran Associates, Inc., 2016, pp. 127–135.

[100] M. Kass, A. Witkin, and D. Terzopoulos. "Snakes: Active Contour Models." In: *International Journal of Computer Vision* 1.4 (Jan. 1988), pp. 321–331.

[101] M. Kass, A. Witkin, and D. Terzopoulos. "A Comparative Study of Modern Inference Techniques for Structured Discrete Energy Minimization Problems." In: *International Journal of Computer Vision* 115.2 (Nov. 2015), pp. 155–184.

[102] B. M. Kelm, M. Wels, S. K. Zhou, S. Seifert, M. Suehling, Y. Zheng, and D. Comaniciu. "Spine detection in CT and MR using iterated marginal space learning." In: *Medical Image Analysis* 17.8 (2013), pp. 1283–1292.

[103] K. Khoshelham. "Extending Generalized Hough Transform to Detect 3d Objects in Laser Range Data." In: *Proceedings of the ISPRS Workshop on Laser Scanning and SilviLaser 2007*, ed. by P. Rönnholm et al. International Society for Photogrammetry and Remote Sensing, 2007, pp. 206–210.

[104] D. P. Kingma and J. L. Ba. *Adam: A Method for Stochastic Optimization*. 2014. arXiv: 1412.6980 [cs.LG].

[105] T. Klinder, J. Ostermann, M. Ehm, A. Franz, R. Kneser, and C. Lorenz. "Automated model-based vertebra detection, identification, and segmentation in CT images." In: *Medical Image Analysis* 13.3 (June 2009), pp. 471–482.

[106] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning. MIT Press, 2009.

[107] N. Komodakis, B. Xiang, and N. Paragios. "A framework for efficient structured max-margin learning of high-order MRF models." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.7 (July 2015), pp. 1425–1441.

[108] Kontschieder, Peter and Kohli, Pushmeet and Shotton, Jamie and Criminisi, Antonio. "GeoF: Geodesic Forests for Learning Coupled Predictors." In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 65–72.

[109] F. Kordon, P. Fischer, M. Privalov, B. Swartman, M. Schnetzke, J. Franke, R. Lasowski, A. Maier, and H. Kunze. "Multi-task Localization and Segmentation for X-Ray Guided Planning in Knee Surgery." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11769. Springer, 2019, pp. 622–630.

[110] P. Krähenbühl and V. Koltun. "Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials." In: *Advances in Neural Information Processing Systems 24*, ed. by J. Shawe-Taylor et al. Curran Associates, Inc., 2011, pp. 109–117.

[111] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger. "Factor Graphs and the Sum-Product Algorithm." In: *IEEE Transactions on Information Theory* 47.2 (Feb. 2001), pp. 498–519.

[112] T. Kurmann, P. Marquez Neila, X. Du, P. Fua, D. Stoyanov, S. Wolf, and R. Sznitman. "Simultaneous Recognition and Pose Estimation of Instruments in Minimally Invasive Surgery." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10434. Springer, 2017, pp. 505–513.

[113] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data." In: *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 2001, pp. 282–289.

[114] I. Laina, N. Rieke, C. Rupprecht, J. P. Vizcaíno, A. Eslami, F. Tombari, and N. Navab. "Concurrent segmentation and localization for tracking of surgical instruments." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10434. Springer, 2017, pp. 664–672.

[115] M. A. Larhmam, S. Mahmoudi, and M. Benjelloun. "Semi-automatic detection of cervical vertebrae in X-ray images using generalized Hough transform." In: *2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA)*. 2012, pp. 396–401.

[116] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning." In: *Nature* 521.7553 (2015), pp. 436–444.

[117] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. J. Huang. "A Tutorial on Energy-Based Learning." In: *Predicting Structured Data*, ed. by G. Bakir et al. MIT Press, 2006.

[118] X. Li, Q. Dou, H. Chen, C.-W. Fu, X. Qi, D. L. Belavý, G. Armbrecht, D. Felsenberg, G. Zheng, and P.-A. Heng. "3D multi-scale FCN with random modality voxel dropout learning for Intervertebral Disc Localization and Segmentation from Multi-modality MR Images." In: *Medical Image Analysis* 45 (Apr. 2018), pp. 41–54.

[119] Y. Li, A. Alansary, J. J. Cerrolaza, B. Khanal, M. Sinclair, J. Matthew, C. Gupta, C. Knight, B. Kainz, and D. Rueckert. "Fast Multiple Landmark Localisation Using a Patch-Based Iterative Network." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11070. Springer, 2018, pp. 563–571.

[120] H. Liao, A. Mesfin, and J. Luo. "Joint Vertebrae Identification and Localization in Spinal CT Images by Combining Short- and Long-Range Contextual Information." In: *IEEE Transactions on Medical Imaging* 37.5 (May 2018), pp. 1266–1275.

[121] M. Lin, Q. Chen, and S. Yan. *Network In Network*. 2013. arXiv: `1312.4400 [cs.NE]`.

[122] C. Lindner, D. Waring, B. Thiruvenkatachari, K. O'Brien, and T. F. Cootes. "Adaptable Landmark Localisation: Applying Model Transfer Learning to a Shape Model Matching System." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10433. Springer, 2017, pp. 144–151.

[123] C. Lindner, P. A. Bromiley, M. C. Ionita, and T. F. Cootes. "Robust and Accurate Shape Model Matching Using Random Forest Regression-Voting." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.9 (July 2015), pp. 1862–1874.

[124] J. Long, E. Shelhamer, and T. Darrell. "Fully Convolutional Networks for Semantic Segmentation." In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 3431–3440.

[125] M. Loog and B. Ginneken. "Segmentation of the posterior ribs in chest radiographs using iterated contextual pixel classification." In: *IEEE Transactions on Medical Imaging* 25.5 (May 2006), pp. 602–611.

[126] C. Lorenz and J. von Berg. "Fast automated object detection by recursive casting of search rays." In: *International Congress Series* 1281 (May 2005), pp. 230–235.

[127] B. C. Lowekamp, D. T. Chen, L. Ibáñez, and D. Blezek. "The design of SimpleITK." In: *Frontiers in Neuroinformatics* 7 (Dec. 2013).

[128] X. Lu, B. Georgescu, M.-P. Jolly, J. Guehring, A. Young, B. Cowan, A. Littmann, and D. Comaniciu. "Cardiac Anchoring in MRI through Context Modeling." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010*, ed. by T. Jiang et al. Lecture Notes in Computer Science 6361. Springer, 2010, pp. 383–390.

[129] W. Luo, Y. Li, R. Urtasun, and R. Zemel. "Understanding the Effective Receptive Field in Deep Convolutional Neural Networks." In: *Advances in Neural Information Processing Systems 29*, ed. by D. D. Lee et al. Curran Associates, Inc., 2016, pp. 4898–4906.

[130] A. O. Mader, J. von Berg, A. Fabritz, C. Lorenz, and C. Meyer. "Localization and Labeling of Posterior Ribs in Chest Radiographs Using a CRF-regularized FCN with Local Refinement." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11071. Springer, 2018, pp. 562–570.

[131] A. O. Mader, J. von Berg, C. Lorenz, and C. Meyer. "A Novel Approach to Handle Inference in Discrete Markov Networks with Large Label Sets." In: *Proceedings of the Ninth International Conference on Probabilistic Graphical Models*, ed. by V. Kratochvíl et al. Proceedings of Machine Learning Research 72. PMLR, 2018, pp. 249–259.

[132] A. O. Mader, C. Lorenz, J. von Berg, and C. Meyer. "Automatically Localizing a Large Set of Spatially Correlated Key Points: A Case Study in Spine Imaging." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11769. Springer, 2019, pp. 384–392.

[133] A. O. Mader, C. Lorenz, M. Bergtholdt, J. von Berg, H. Schramm, J. Modersitzki, and C. Meyer. "Detection and Localization of Landmarks in the Lower Extremities Using an Automatically Learned Conditional Random Field." In: *Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics*, ed. by M. J. Cardoso et al. Lecture Notes in Computer Science 10551. Springer, 2017, pp. 64–75.

[134] A. O. Mader, C. Lorenz, M. Bergtholdt, J. von Berg, H. Schramm, J. Modersitzki, and C. Meyer. "Detection and localization of spatially correlated point landmarks in medical images using an automatically learned conditional random field." In: *Computer Vision and Image Understanding* 176–177 (Dec. 2018), ed. by N. Paragios et al., pp. 45–53.

[135] A. O. Mader, C. Lorenz, and C. Meyer. "A General Framework for Localizing and Locally Segmenting Correlated Objects: A Case Study on Intervertebral Discs in Multi-Modality MR Images." In: *Medical Image Understanding and Analysis*, ed. by Y. Zheng et al. Communications in Computer and Information Science 1065. Springer, 2020, pp. 364–376.

[136] A. O. Mader, H. Schramm, and C. Meyer. "Efficient Epiphyses Localization Using Regression Tree Ensembles and a Conditional Random Field." In: *Bildverarbeitung für die Medizin 2017*, ed. by K. H. Maier-Hein et al. Informatik aktuell. Springer, 2017, pp. 179–184.

[137] K. V. Mardia and P. E. Jupp. *Directional Statistics*. Wiley, 1999.

[138] M. M. McCormick, X. Liu, L. Ibanez, J. Jomier, and C. Marion. "ITK: enabling reproducible research and open science." In: *Frontiers in Neuroinformatics* 8 (Feb. 2014).

[139] 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.

[140] M. Monteiro, M. A. T. Figueiredo, and A. L. Oliveira. *Conditional Random Fields as Recurrent Neural Networks for 3D Medical Imaging Segmentation*. 2018. arXiv: 1807.07464 [cs.CV].

[141] Y. Mrabet. *Human anatomy planes*. 2008. Available at: https://commons.wikimedia.org/wiki/File:Human_anatomy_planes.svg (visited on 04/06/2020).

[142] C. Mwikirize, J. L. Nosher, and I. Hacihaliloglu. "Single Shot Needle Tip Localization in 2D Ultrasound." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11768. Springer, 2019, pp. 637–645.

[143] A. Nguyen, J. Yosinski, and J. Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 427–436.

[144] L. G. Nyúl, J. K. Udupa, and X. Zhang. "New variants of a method of MRI scale standardization." In: *IEEE Transactions on Medical Imaging* 19.2 (Feb. 2000), pp. 143–150.

[145] M. Oda, N. Shimizu, H. R. Roth, K. Karasawa, T. Kitasaka, K. Misawa, M. Fujiwara, D. Rueckert, and K. Mori. "3D FCN Feature Driven Regression Forest-Based Pancreas Localization and Segmentation." In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, ed. by M. J. Cardoso et al. Lecture Notes in Computer Science 10553. Springer, 2017, pp. 222–230.

[146] T. Ojala, M. Pietikäinen, and D. Harwood. "A comparative study of texture measures with classification based on featured distributions." In: *Pattern recognition* 29.1 (1996), pp. 51–59.

[147] R. Okada. "Discriminative Generalized Hough Transform for Object Dectection." In: *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 2000–2005.

[148] O. Oktay, W. Bai, R. Guerrero, M. Rajchl, A. de Marvao, D. P. O'Regan, S. A. Cook, M. P. Heinrich, B. Glocker, and D. Rueckert. "Stratified Decision Forests for Accurate Anatomical Landmark Localization in Cardiac Images." In: *IEEE Transactions on Medical Imaging* 36.1 (Jan. 2017), pp. 332–342.

[149] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. "Towards Accurate Multi-person Pose Estimation in the Wild." In: *2017 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017, pp. 3711–3719.

[150] C. Payer, D. Štern, H. Bischof, and M. Urschler. "Regressing Heatmaps for Multiple Landmark Localization Using CNNs." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, ed. by S. Ourselin et al. Lecture Notes in Computer Science 4584. Springer, 2016, pp. 230–238.

[151] C. Payer, D. Štern, H. Bischof, and M. Urschler. "Integrating spatial configuration into heatmap regression based CNNs for landmark localization." In: *Medical Image Analysis* 54 (May 2019), pp. 207–219.

[152] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. "Scikit-learn: Machine learning in Python." In: *Journal of Machine Learning Research* 12.85 (Dec. 2011), pp. 2825–2830.

[153] M. P. Perrone. "Improving regression estimation: Averaging methods for variance reduction with extensions to general convex measure optimization." PhD thesis. Brown University, 1993.

[154] T. Pfister, J. Charles, and A. Zisserman. "Flowing ConvNets for Human Pose Estimation in Videos." In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 1913–1921.

[155] T. V. Pham and A. W. M. Smeulders. "Object recognition with uncertain geometry and uncertain part detection." In: *Computer Vision and Image Understanding* 99.2 (Aug. 2005), pp. 241–258.

[156] E. Pietka, A. Gertych, S. Pospiech, F. Cao, H. Huang, and V. Gilsanz. "Computer-Assisted Bone Age Assessment: Image Preprocessing and Epiphyseal/Metaphyseal ROI Extraction." In: *IEEE Transactions on Medical Imaging* 20.8 (Aug. 2001), pp. 715–729.

[157] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. "Poselet Conditioned Pictorial Structures." In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 588–595.

[158] C. Qian, L. Wang, Y. Gao, A. Yousuf, X. Yang, A. Oto, and D. Shen. "In vivo MRI based prostate cancer localization with random forests and auto-context model." In: *Computerized Medical Imaging and Graphics* 52 (Sept. 2016), pp. 44–57.

[159] J. R. Quinlan. "Induction of decision trees." In: *Machine Learning* 1.1 (Mar. 1986), pp. 81–106.

[160] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1992.

[161] D. Richmond, D. Kainmueller, B. Glocker, C. Rother, and G. Myers. "Uncertainty-Driven Forest Predictors for Vertebra Localization and Segmentation." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, ed. by N. Navab et al. Lecture Notes in Computer Science 9349. Springer, 2015, pp. 653–660.

[162] D. L. Richmond, D. Kainmueller, M. Y. Yang, E. W. Myers, and C. Rother. *Mapping Auto-context Decision Forests to Deep ConvNets for Semantic Segmentation*. 2018. arXiv: 1507.07583v3 [cs.CV].

[163] J. Richter. "Investigations on Structure Learning in Markov Random Fields." MA thesis. University of Applied Sciences Kiel, 2017.

[164] O. Ronneberger, P. Fischer, and T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, ed. by N. Navab et al. Lecture Notes in Computer Science 9351. Springer, 2015, pp. 234–241.

[165] E. Rosten and T. Drummond. "Machine Learning for High-Speed Corner Detection." In: *Computer Vision – ECCV 2006*, ed. by A. Leonardis et al. Springer, 2006, pp. 430–443.

[166] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors." In: *Nature* 323.6088 (Oct. 1986), pp. 533–536.

[167] H. Ruppertshofen. "Automatic Modeling of Anatomical Variability for Object Localization in Medical Images." PhD thesis. Otto-von-Guericke-Universität Magdeburg, 2013.

[168] H. Ruppertshofen, C. Lorenz, S. Schmidt, P. Beyerlein, Z. Salah, G. Rose, and H. Schramm. "Discriminative Generalized Hough transform for localization of joints in the lower extremities." In: *Computer Science - Research and Development* 26.1 (Feb. 2011), pp. 97–105.

[169] S. S. M. Salehi, D. Erdogmus, and A. Gholipour. "Auto-Context Convolutional Neural Network (Auto-Net) for Brain Extraction in Magnetic Resonance Imaging." In: *IEEE Transactions on Medical Imaging* 36.11 (Nov. 2017), pp. 2319–2330.

[170] W. Samek and K.-R. Müller. "Towards explainable artificial intelligence." In: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, ed. by W. Samek et al. Lecture Notes in Computer Science 11700. Springer, 2019, pp. 5–22.

[171] F. Schlüter, F. Bromberg, and A. Edera. "The IBMAP approach for Markov network structure learning." In: *Annals of Mathematics and Artificial Intelligence* 72.3 (Nov. 2014), pp. 197–223.

[172] S. Schmidt, J. Kappes, M. Bergtholdt, V. Pekar, S. Dries, D. Bystrov, and C. Schnörr. "Spine Detection and Labeling Using a Parts-Based Graphical Model." In: *Information Processing in Medical Imaging*, ed. by N. Karssemeijer et al. Lecture Notes in Computer Science 4584. Springer, 2007, pp. 122–133.

[173] H. Schramm, O. Ecabert, J. Peters, V. Philomin, and J. Weese. "Toward fully automatic object detection and segmentation." In: *Medical Imaging 2006: Image Processing*, ed. by J. M. Reinhardt et al. Vol. 6144. SPIE, 2006, pp. 11–20.

[174] A. Sekuboyina, A. Bayat, M. E. Husseini, M. Löffler, M. Rempfler, J. Kukačka, G. Tetteh, A. Valentinitsch, C. Payer, M. Urschler, et al. *VerSe: A Vertebrae Labelling and Segmentation Benchmark*. Jan. 2020. arXiv: `2001.09193 [cs.CV]`.

[175] A. Sekuboyina, M. Rempfler, J. Kukačka, G. Tetteh, A. Valentinitsch, J. S. Kirschke, and B. H. Menze. "Btrfly Net: Vertebrae Labelling with Energy-Based Adversarial Learning of Local Spine Prior." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11073. Springer, 2018, pp. 649–657.

[176] J. Sénégas, A. Saalbach, M. Bergtholdt, S. Jockel, D. Mentrup, and R. Fischbach. "Evaluation of Collimation Prediction Based on Depth Images and Automated Landmark Detection for Routine Clinical Chest X-Ray Exams." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11071. Springer, 2018, pp. 571–579.

[177] M. P. Shah, S. N. Merchant, and S. P. Awate. "MS-Net: Mixed-Supervision Fully-Convolutional Networks for Full-Resolution Segmentation." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11073. Springer, 2018, pp. 379–387.

[178] T. Sharp. "Implementing Decision Trees and Forests on a GPU." In: *Computer Vision – ECCV 2008*, ed. by D. Forsyth et al. Lecture Notes in Computer Science 5305. Springer, 2008, pp. 595–608.

[179] C. Shen, J. He, Y. Huang, and J. Wu. "Discriminative Correlation Filter Network for Robust Landmark Tracking in Ultrasound Guided Intervention." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11768. Springer, 2019, pp. 646–654.

[180] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning." In: *IEEE Transactions on Medical Imaging* 35.5 (May 2016), pp. 1285–1298.

[181] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, et al. "Efficient Human Pose Estimation from Single Depth Images." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12 (Dec. 2012), pp. 2821–2840.

[182] P. Y. Simard, D. Steinkraus, and J. C. Platt. "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis." In: *Proceedings of the Seventh International Conference on Document Analysis and Recognition*. Vol. 2. IEEE, 2003, pp. 958–963.

[183] M. Sofka, F. Milletari, J. Jia, and A. Rothberg. "Fully Convolutional Regression Network for Accurate Detection of Measurement Points." In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, ed. by M. J. Cardoso et al. Lecture Notes in Computer Science 10553. Springer, 2017, pp. 258–266.

[184] D. Štern, T. Ebner, and M. Urschler. "From Local to Global Random Regression Forests: Exploring Anatomical Landmark Localization." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, ed. by S. Ourselin et al. Lecture Notes in Computer Science 9901. Springer, 2016, pp. 221–229.

[185] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. Jorge Cardoso. "Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations." In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, ed. by M. J. Cardoso et al. Springer, 2017, pp. 240–248.

[186] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. "Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation." In: *Advances in Neural Information Processing*

Systems 27, ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 1799–1807.

[187] G. J. Tortora and B. H. Derrickson. *Principles of Anatomy and Physiology*. 15th ed. Wiley, 2016.

[188] A. Toshev and C. Szegedy. "DeepPose: Human Pose Estimation via Deep Neural Networks." In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1653–1660.

[189] N. Toussaint, B. Khanal, M. Sinclair, A. Gomez, E. Skelton, J. Matthew, and J. A. Schnabel. "Weakly Supervised Localisation for Fetal Ultrasound Images." In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, ed. by D. Stoyanov et al. Lecture Notes in Computer Science 11045. Springer, 2018, pp. 192–200.

[190] Z. Tu. "Probabilistic Boosting-Tree: Learning Discriminative Models for Classification, Recognition, and Clustering." In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. IEEE, 2005, pp. 1589–1596.

[191] Z. Tu. "Auto-context and Its Application to High-level Vision Tasks." In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[192] Z. Tu and X. Bai. "Auto-Context and Its Application to High-Level Vision Tasks and 3D Brain Image Segmentation." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.10 (Oct. 2009), pp. 1744–1757.

[193] A. Tuysuzoglu, J. Tan, K. Eissa, A. P. Kiraly, M. Diallo, and A. Kamen. "Deep Adversarial Context-Aware Landmark Detection for Ultrasound Imaging." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11073. Springer, 2018, pp. 151–158.

[194] M. Ulrich, C. Steger, and A. Baumgartner. "Real-time object recognition using a modified generalized Hough transform." In: *Pattern Recognition* 36.11 (Nov. 2003), pp. 2557–2570.

[195] M. Urschler, T. Ebner, and D. Štern. "Integrating geometric configuration and appearance information into a unified framework for anatomical landmark localization." In: *Medical Image Analysis* 43 (Jan. 2018), pp. 23–36.

[196] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al. "SciPy 1.0: fundamental algorithms for scientific computing in Python." In: *Nature Methods* (Mar. 2020), pp. 261–272.

[197] A. Vlontzos, A. Alansary, K. Kamnitsas, D. Rueckert, and B. Kainz. "Multiple Landmark Detection using Multi-Agent Reinforcement Learning." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11767. Springer, 2019, pp. 565–572.

[198] K. Vogt, O. Müller, and J. Ostermann. "Facial Landmark Localization Using Robust Relationship Priors and Approximative Gibbs Sampling." In: *Advances in Visual Computing*, ed. by G. Bebis et al. Lecture Notes in Computer Science 9475. Springer, 2015, pp. 365–376.

[199] S. v. d. Walt, S. C. Colbert, and G. Varoquaux. "The NumPy Array: A Structure for Efficient Numerical Computation." In: *Computing in Science & Engineering* 13.2 (Mar. 2011), pp. 22–30.

[200] C. Wang, N. Komodakis, and N. Paragios. "Markov Random Field modeling, inference & learning in computer vision & image understanding: A survey." In: *Computer Vision and Image Understanding* 117.11 (2013), pp. 1610–1627.

[201] A. G. Webb. *Introduction to Biomedical Imaging*. Wiley, 2003.

[202] M. Weber. "Unsupervised Learning of Models for Object Recognition." PhD thesis. California Institute of Technology, 2000.

[203] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. "Convolutional Pose Machines." In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 4724–4732.

[204] J. Wessel, M. P. Heinrich, J. von Berg, A. Franz, and A. Saalbach. *Sequential Rib Labeling and Segmentation in Chest X-Ray using Mask R-CNN*. 2019. arXiv: 1908.08329 [eess.IV].

[205] D. Wu, D. Liu, Z. Puskas, C. Lu, A. Wimmer, C. Tietjen, G. Soza, and S. K. Zhou. "A learning based deformable template matching method for automatic rib centerline extraction and labeling in CT images." In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 980–987.

[206] H. Wu, C. Bailey, P. Rasoulinejad, and S. Li. "Automatic Landmark Estimation for Adolescent Idiopathic Scoliosis Assessment Using BoostNet." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10433. Springer, 2017, pp. 127–135.

[207] Z. Xu, Q. Huang, J. Park, M. Chen, D. Xu, D. Yang, D. Liu, and S. K. Zhou. "Supervised Action Classifier: Approaching Landmark Detection as Image Partitioning." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10435. Springer, 2017, pp. 338–346.

[208] Z. Xu, Y. Huo, J. Park, B. Landman, A. Milkowski, S. Grbic, and S. Zhou. "Less is More: Simultaneous View Classification and Landmark Detection for Abdominal Ultrasound Images." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2018*, ed. by A. F. Frangi et al. Lecture Notes in Computer Science 11071. Springer, 2018, pp. 711–719.

[209] D. Yang, T. Xiong, D. Xu, S. K. Zhou, Z. Xu, M. Chen, J. Park, S. Grbic, T. D. Tran, S. P. Chin, et al. "Deep Image-to-Image Recurrent Network with Shape Basis Learning for Automatic Vertebra Labeling in Large-Scale 3D CT Volumes." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017*, ed. by M. Descoteaux et al. Lecture Notes in Computer Science 10435. Springer, 2017, pp. 498–506.

[210] X. Yang, W. Shi, H. Dou, J. Qian, Y. Wang, W. Xue, S. Li, D. Ni, and P.-A. Heng. "FetusMap: Fetal Pose Estimation in 3D Ultrasound." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11768. Springer, 2019, pp. 281–289.

[211] Y. Yang and D. Ramanan. "Articulated Human Detection with Flexible Mixtures of Parts." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12 (Dec. 2013), pp. 2878–2890.

[212] J. C. Ye and W. K. Sung. "Understanding Geometry of Encoder-Decoder CNNs." In: *Proceedings of the 36th International Conference on Machine Learning*, ed. by K. Chaudhuri et al. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 7064–7073.

[213] F. Yu and V. Koltun. *Multi-Scale Context Aggregation by Dilated Convolutions*. 2015. arXiv: `1511.07122 [cs.CV]`.

[214] F. Zana and J.-C. Klein. "A multimodal registration algorithm of eye fundus images using vessels detection and Hough transform." In: *IEEE Transactions on Medical Imaging* 18.5 (May 1999), pp. 419–428.

[215] G. Zeng, D. Belavy, S. Li, and G. Zheng. "Evaluation and Comparison of Automatic Intervertebral Disc Localization and Segmentation methods with 3D Multi-modality MR Images: A Grand Challenge." In: *Computational Methods and Clinical Applications for Spine Imaging*, ed. by G. Zheng et al. Lecture Notes in Computer Science 11397. Springer, 2019, pp. 163–171.

[216] J. Zhang, M. Liu, and D. Shen. "Detecting Anatomical Landmarks From Limited Medical Imaging Data Using Two-Stage Task-Oriented Deep Neural Networks." In: *IEEE Transactions on Image Processing* 26.10 (Oct. 2017), pp. 4753–4764.

[217] Y. Zhang and T. Chen. "Implicit Shape Kernel for Discriminative Learning of the Hough Transform Detector." In: *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 105.1–105.11.

[218] C. Zhao, J. Wang, G. Zhu, Y. Wu, and H. Lu. "Learning weighted part models for object tracking." In: *Computer Vision and Image Understanding* 143 (Feb. 2016), pp. 173–182.

[219] S. Zhao, X. Wu, B. Chen, and S. Li. "Automatic Vertebrae Recognition from Arbitrary Spine MRI Images by a Hierarchical Self-calibration Detection Framework." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11767. Springer, 2019, pp. 316–325.

[220] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr. "Conditional Random Fields as Recurrent Neural Networks." In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 1529–1537.

[221] Y. Zheng, B. Georgescu, and D. Comaniciu. "Marginal Space Learning for Efficient Detection of 2D/3D Anatomical Structures in Medical Images." In: *Information Processing in Medical Imaging*, ed. by J. L. Prince et al. Lecture Notes in Computer Science 5636. Springer, 2009, pp. 411–422.

[222] Y. Zheng, D. Liu, B. Georgescu, H. Nguyen, and D. Comaniciu. "3D Deep Learning for Efficient and Robust Landmark Detection in Volumetric Data." In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, ed. by N. Navab et al. Lecture Notes in Computer Science 9349. Springer, 2015, pp. 565–572.

[223] Z. Zhong, J. Li, Z. Zhang, Z. Jiao, and X. Gao. "An Attention-Guided Deep Regression Model for Landmark Detection in Cephalograms." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, ed. by D. Shen et al. Lecture Notes in Computer Science 11769. Springer, 2019, pp. 540–548.

[224] D. Ziou and S. Tabbone. "Edge Detection Techniques - An Overview." In: *Pattern Recognition & Image Analysis* 8.4 (1998), pp. 537–559.

Part V

APPENDIX

A

JOINT OPTIMIZATION OF WEIGHTS AND POTENTIALS

Currently, the parameters of the potential functions and the combining CRF parameters weights and "missing" energies are estimated independently. However, we also acknowledged in Section 7.2 that a joint optimization of both parts is possible if the potential functions are derivable w. r. t. to their parameters. Such a joint optimization might provide better results than an independent optimization. However, not all potential functions that we used are derivable w. r. t. their parameters. Thus, to illustrate the potential of such an optimization we set out to create a proof-of-concept experiment to illustrate the general idea.

A.1 DERIVABLE POTENTIAL FUNCTIONS

First of all, we define a unary and a binary potential function that are derivable, easy to compute and should provide reasonable results w. r. t. the lessons learned.

A.1.1 *Unary CNN Potential*

We already see that CNNs—which are easily derivable—provide great accuracy despite a potential downsampling in the output domain, but failed to provide reasonable alternatives in case of failure. We suggested that this problem is related to an inappropriate receptive field, which covers more than just the object of importance. Here, we use a very simple CNN architecture tailored towards the used dataset while trying to minimize the overall memory consumption.

The architecture in form of the layers—only convolutional layers have been used—is listed in Table A.1. Note that we zero-pad the input feature maps prior to any convolution and use the ReLU activation function for all layers except the last one. The resulting receptive field size of $45 \times 45 \times 45$ vx is sufficiently large to capture the object of interest (vertebra, as explained later), but not too large in order to focus only on the object rather than the composition of objects. The reduced output size caused by the downsampling with factor 4 and the resulting highly quantized localization hypotheses are again countered by the polynomial refinement as explained in Section 8.3. Note that we used two layers with dilated convolutions[1] to further increase the receptive field size without loss of resolution.

[1] We refer the reader to [213] for an introduction to dilated convolutions.

Table A.1: Listing of the layers—only convolutional—forming our simple dataset-specific CNN in terms of isotropic kernel size, number of output feature maps, stride and dilation. Note that a ReLU activation function is used after each layer except the last one. This architecture results in an isotropic receptive field size of 45^3 vx and a downsampling factor of 4. The "-" indicates a default value of 1.

LAYER	KERNEL SIZE	FEATURE MAPS	STRIDE	DILATION
1	3	64	-	-
2	3	64	2	-
3	3	128	-	-
4	3	128	2	-
5	3	256	-	2
6	3	256	-	2
7	1	N	-	-

In order to create a probabilistically-scaled network output, we use the Gibbs measure[2] to incorporate all voxel outputs when applying the point-based loss formulation, which is explained later. The unary potential is then given by

$$\psi_i(\boldsymbol{x}_i\,;\,\boldsymbol{\Theta}_i,\boldsymbol{I}) = -\log\frac{\exp(\boldsymbol{Y}_i(\boldsymbol{x}_i\,;\,\boldsymbol{\Theta}_i))}{\sum_{\boldsymbol{x}'\in\boldsymbol{I}}\exp(\boldsymbol{Y}_i(\boldsymbol{x}'\,;\,\boldsymbol{\Theta}_i))}\,, \tag{A.1}$$

with $\boldsymbol{Y}_i$ being the network output after the last layer w. r. t. the network parameters $\boldsymbol{\Theta}_i$ (i. e., weights and biases).

A.1.2 *Binary Frequented Vector Potential*

For constellation modelling, we use a histogram-based binary vector—recall Eq. (6.14)—potential in order to not make any assumptions about the underlying distribution, but to have a nice visual interpretability. Additionally, it is very easy to formulate and optimize.

Hence, the binary potential function is given by

$$\psi_{i,j}(\boldsymbol{x}_i,\boldsymbol{x}_j\,;\,\boldsymbol{\Theta}_{i,j}) = -\log\frac{\exp\left(\mathrm{TriLinInt}(\boldsymbol{\Theta}_{i,j},\boldsymbol{v})\right)}{\sum_{\boldsymbol{v}'\in\boldsymbol{\Theta}_{i,j}}\exp(\boldsymbol{\Theta}_{i,j}[\boldsymbol{v}'])}\,, \tag{A.2}$$

with $\boldsymbol{v} = \boldsymbol{x}_i - \boldsymbol{x}_j$ being the displacement vector. Note that instead of using the value $\boldsymbol{\Theta}_{i,j}[\boldsymbol{v}']$ at the quantized position $\boldsymbol{v}' = \lfloor \boldsymbol{v} + 0.5 \rfloor$ directly, we use trilinear interpolation to provide reasonable values at off-grid vectors by incorporating values from the corresponding 2×2×2 vx cube rather than just one value. For brevity, the trilinear interpolation is here just indicated by "TriLinInt". Remember that the off-grid vectors are

[2] Note that in practice the log-sum-exp trick has to be used in order to prevent overflow/underflow issues, which essentially subtracts the maximum from each component.

caused by the polynomial refinement of the localization hypotheses, which can have a more severe effect if the images have a low resolution (which is here the case, explained later).

A.2 ADJUSTED LOSS FORMULATION

Given those potential functions, the loss formulation from Eq. (7.1) can additionally be partially derived w. r. t. the potential function parameters $\Theta = \{\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_{1,2}, \boldsymbol{\Theta}_2, ...\}$. However, this is not sufficient since the unary potentials— from which the set of localization hypotheses is extracted including the correct configuration $\boldsymbol{s}_k^+$ and the best matching incorrect configuration $\boldsymbol{s}_k^-$—are not trained in the beginning and hence provide no correct localization hypotheses and thus no reasonable correct rival $\boldsymbol{s}_k^+$. To overcome this problem, an additional loss term is introduced which optimizes the potentials individually using the annotated positions $\overset{\star}{\boldsymbol{x}}_i$. This also has the added benefit, that the potential functions are further optimized in case the max-filtered state term is already satisfied but which might have been caused by other potential functions, which in turn would prevent the potential function from learning. Note that we use the annotated positions $\overset{\star}{\boldsymbol{x}}_i$ to construct the correct rival $\boldsymbol{s}_k^+$ for the state loss term in case no correct rival can be constructed from the set of localization hypotheses, which is mostly the case in the beginning of the optimization. To this end, the loss function on K training samples $k \in [1\,..\,K]$ is defined as

$$
\begin{aligned}
L(\boldsymbol{\Lambda}, \Theta) = \frac{1}{K} \sum_{k=1}^{K} \Bigg[& \max\big(0, m + E(\boldsymbol{s}_k^+ \mid \boldsymbol{\Lambda}, \Theta) \\
& \qquad - E(\boldsymbol{s}_k^- \mid \boldsymbol{\Lambda}, \Theta)\big) \\
& + \sum_{f=1}^{F} \max\big(0, \frac{m}{F} + \psi_f(\overset{\star}{\boldsymbol{x}}^k_{c_{f,1}}, \ldots, \overset{\star}{\boldsymbol{x}}^k_{c_{f,C_f}}; \boldsymbol{\Theta}_f) \\
& \qquad - \psi_f(\boldsymbol{x}_k^-(c_{f,1}), \ldots, \boldsymbol{x}_k^-(c_{f,C_f}); \boldsymbol{\Theta}_f)\big) \Bigg].
\end{aligned}
\tag{A.3}
$$

The same notation as in Section 7.2 has been used, with $\boldsymbol{x}_k^-(i) = \boldsymbol{x}_{i,\boldsymbol{s}^-_{k,i}}$ selecting the localization hypothesis of the given key point w. r. t. the incorrect rival $\boldsymbol{s}_k^-$.

Note that "missing" energies are not optimized in this formulation, since the dataset used in our experiment (explained later) does not pose the detection problem. However, they are easily integrated by analogously applying the case filtering from Eq. (4.1) in our added potential loss term.

A.3 EXPERIMENTAL SETUP

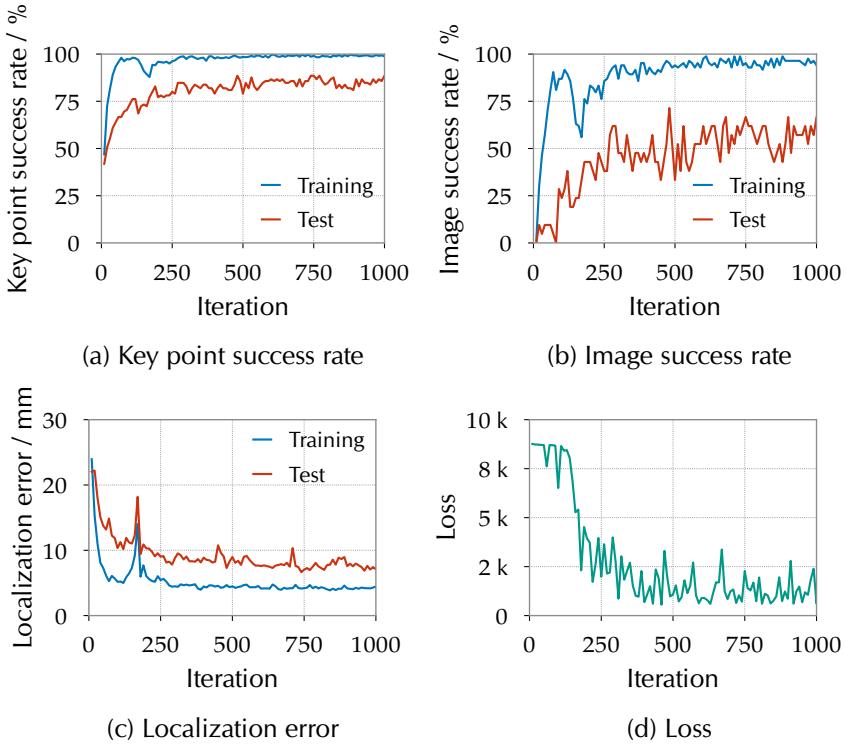
Due to the comparably high memory demand, a very simplistic experiment has been conducted to illustrate the feasibility. The full spines CT scans were resampled to an isotropic resolution of $3\times3\times3\,\mathrm{mm}^3/\mathrm{vx}$ and images that contained transformed annotations at off-grid boundary positions (e. g., if any axis position is in $[-0.5, 0)$) were excluded, resulting in 105 images. Of those images, 80 % were used for training (84) while the remaining 20 % were used for testing (21). In order to further reduce the memory demand, only the centroids of the 5 lumbar vertebrae have been used as target key points.

The optimization was performed from scratch, i. e., without pretraining any of the potential functions. The binary potential functions were placed in form of a chain along the spinal column, resulting in 4 binary and 5 unary potential functions. Again, we used SGD in form of the Adam algorithm with a learning rate of 1E−4 and a mini-batch size of 8 images. To further accelerate the training, we used the 10 best rivals (i. e., incorrect states with lowest energy) instead of just one.

A.4 RESULTS

The course of the optimization over 1000 iterations is depicted in Fig. A.1. First, we can see that the model successfully learns to localize the key points starting from unoptimized potential functions. Second, the capacity of the model is large enough to solve that localization task, which is evident in the near perfect training key point success rate of 98.8 % with an average localization error of 4.48 mm. Third, the model also generalizes to the unseen test data with a key point success rate of 88.6 % and an average localization error of 7.1 mm. This is especially astonishing, since the CNN has an output resolution of $12\times12\times12\,\mathrm{mm}^3/\mathrm{vx}$, which is only compensated by the polynomial refinement. Additionally, due to the object-specific receptive field size of $13.5\times13.5\times13.5\,\mathrm{cm}^3$, the CNN is capable of providing reasonable alternative localization hypotheses, which manifests in a theoretical (cheating) test key point success rate of 99.1 %.

How the potential functions change over the course of the optimization is depicted in Fig. A.2 in form of the energies computed for two unary as well as two binary potential functions given a random test image. One can see that each potential function indeed learns to properly pinpoint the target key point. Note that the reduced accuracy of the CNN is evident in the coarse display of the energies of the unary potentials.



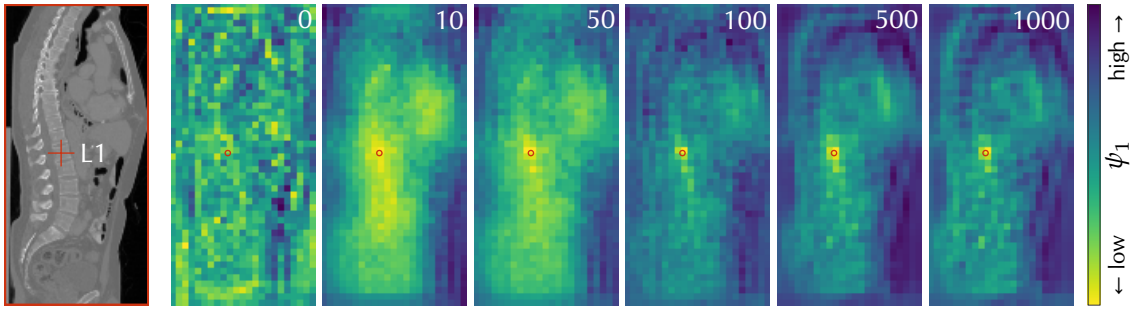
(a) Key point success rate

(b) Image success rate

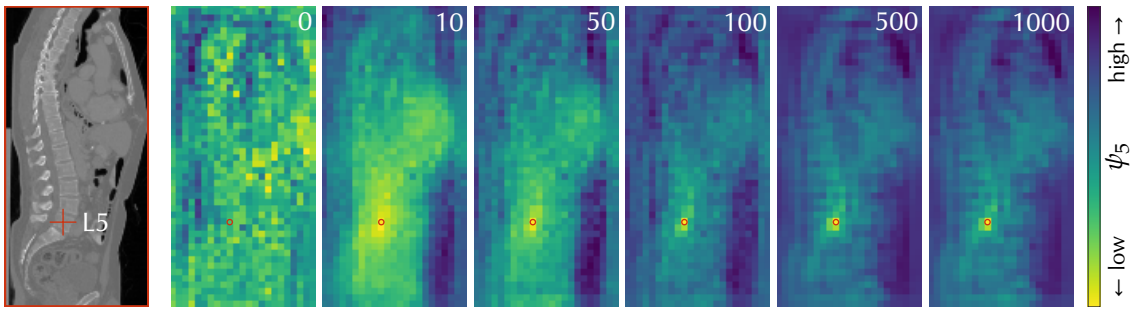
(c) Localization error

(d) Loss

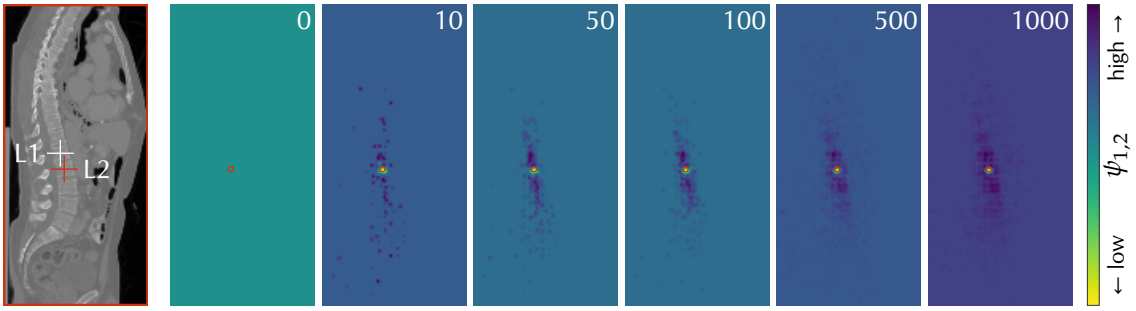
Figure A.1: Training and test performance as a function of the number of iterations in terms of the (a) key point success rate in percent, the (b) image success rate in percent and the (c) localization error in mm. The loss L computed over the training mini-batches is shown in (d).



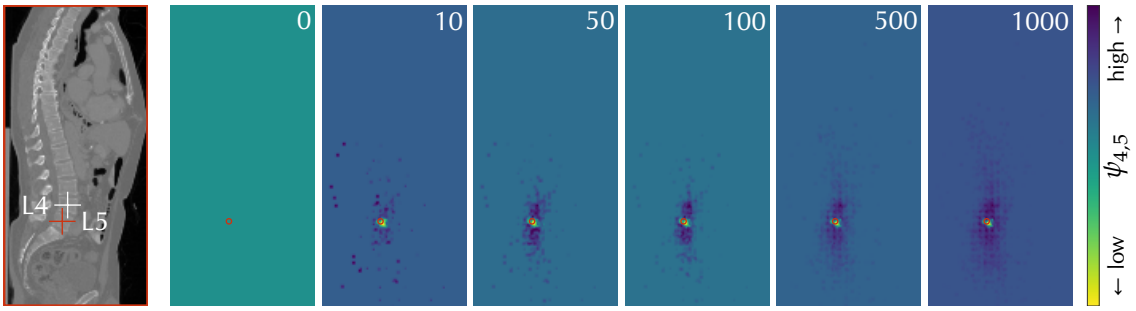
(a) Unary potential function of L1



(b) Unary potential function of L5



(c) Binary potential function of L1 and L2



(d) Binary potential function of L4 and L5

Figure A.2: Illustration of energies computed for some exemplary potential functions and a given test image over the course of the optimization shown as mid-sagittal slices. The unary potential function (i. e., the CNN) is depicted in (a) and (b) for the associated key points L1 and L5, while the binary potential function (vector frequency) is depicted in (c) and (d) for the key point pairs L1–L2 and L4–L5, respectively. For each potential function, the energies computed in the iterations 0, 10, 50, 100, 500 and 1000 are shown. Note that the position of the first key point in case of binary potential functions was fixed to the annotation of that key point, while varying the position of the second key point. The annotated position of the target key point is shown as ■ "+" and ■ circle, where the center of the circle is the annotated position and the area enclosed by it corresponds to the area where a prediction is treated as correctly localized.

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre hiermit, dass es sich bei der eingereichten Arbeit um mein eigenständig erstelltes Werk handelt, welches den Regeln guter wissenschaftlicher Praxis entspricht.

Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtliche und nicht wörtliche Zitate aus anderen Werken - ebenso wie den Ursprung von abgeleiteten Grafiken - als solche kenntlich gemacht.

Die Arbeit oder Teile davon habe ich bislang nicht an einer anderen Stelle im Rahmen eines Prüfungsverfahrens vorgelegt. Die bereits in Papieren publizierten wissenschaftlichen Ergebnisse wurden entsprechend kenntlich gemacht. Darüber hinaus wurde mir kein akademischer Grad entzogen.