Automated Distinct Bone Segmentation from Computed Tomography Images using Deep Learning

Inaugural dissertation

to
be awarded the degree of

Dr. sc. med.

presented at
the Faculty of Medicine
of the University of Basel

by
Eva Schnider
from Breitenbach, Switzerland

Basel, 2023

Originaldokument gespeichert auf dem Dokumentenserver der Universität Basel
edoc.unibas.ch

Approved by the Faculty of Medicine
on application of

Prof. Dr. Philippe C. Cattin, University of Basel – *primary advisor*
PD Dr. med. Gregory Jost, University of Basel, Spitalzentrum Biel – *secondary advisor*
Prof. Dr. Elin Trägårdh, Lund University – *external expert*
Dr. Antal Huck, University of Basel – *further advisor*

Basel, the 16th of January 2023

Prof. Dr. Primo Schär
Dean

They told me computers could only do arithmetic.
Grace Hopper

Contents

Acronyms

1D one dimensional. 19, 22, 25

2D two dimensional. 19, 20, 26, 41, 91

3D three dimensional. 1, 13, 15, 19, 20, 24, 37, 38, 41, 77, 91

AI artificial intelligence. viii, 18

ANN artificial neural network. 17–19, 23

AR augmented reality. 1, 15, 92

CNN convolutional neural network. 19–21, 24, 36, 37, 91

CT Computed Tomography. ix, 1, 2, 5, 11, 19, 21, 36, 37, 91–94

DBE Department of Biomedical Engineering. vii

DSC Dice similarity coefficient. 28, 33, 93

HD Hausdorff distance. 34

HU Hounsfield Unit. 11, 35

MICCAI the International Conference on Medical Image Computing and Computer Assisted Intervention. 41

MIRACLE Minimally-Invasive Robot-Assisted Computer-guided LaserosteotomE. viii, 1, 92

ML machine learning. 17, 18

MRI magnetic resonance imaging. 11, 19, 92

PET positron emission tomography. 37

ReLU rectified linear unit. 23

SVM support vector machine. 36

VR virtual reality. ix, 1, 2, 14, 15, 92

Acknowledgements

This thesis is not the result of solitary work but of work conducted among a network of colleagues, mentors, and friends, whom I would like to thank profoundly. First and foremost, I want to thank my principal advisor, Prof. Dr. Philippe Cattin, for providing me with the opportunity to conduct this Ph.D. and for the guidance, encouragement, and support all the way. I am also very grateful for his openness to me pursuing other projects in parallel to this Ph.D. I further want to thank Dr. Antal Huck, who always found the time to provide thorough and helpful feedback on any publication and to discuss possibilities and options when things did not work out the way they should have. A big thank-you also goes to Dr. Uri Nahum who initially brought me to the Department of Biomedical Engineering (DBE) for a Master's thesis and who fuelled my enjoyment of numerical analysis. For their co-authorship and their valued suggestions, I want to thank Prof. Dr. Georg Rauter and Prof. Dr. Azhar Zam; as well as Prof. Dr. Magdalena Müller-Gerbl and Mireille Toranelli from the Institute of Anatomy who provided the data this work is built on.

As proven by almost one and a half years of Corona-induced working from home, a Ph.D. without colleagues is just not the same. Therefore I would like to thank all of the wonderful people at DBE whom I had the pleasure to work and eat cake with: Carlo Seppi, Balázs Faludi, Prof. Dr. Lilian Witthauer, Dr. Lorenzo Iafolla, Madina Kojanazarova, Marek Żelechowski, Massimiliano Filipozzi, Norbert Zentai, Samaneh Manavi, and Dr. Sara Freund, for the great Navigators vibe. It was really fun to share an office with you! Florentin Bieder, and Julia Wolleb, for the great times in summer schools and for having some of my fellow mathematicians around. Alicia Durrer, Dr. Alina Giger, Dr. Christoph Jud, Nair von Mühlenen, Peter von Niederhäusern, Philippe Valmaggia, and Dr. Robin Sandkühler, from the CIAN group, for making trips to the 4th floor always worthwhile. Cédric Duverney, Esther Zoller, Lorin Fasel, Dr. Manuela Eugster, Mohammad Kair Nahhas, Dr. Nicolas Gerig, and Murali Karnam from the BIROMED lab for the fun during lunch and Rhine swimming, and for your generosity in sharing meeting desserts. Yukiko Tomooka for our fun lunches that improved my Japanese tremendously: お疲れさまでした! Bruno Sempéré, Dr. Ludovic Amruthalingam, Prof. Dr. Mathieu Sarracanie, and Prof. Dr. Najat Salameh, who along with many CIAN members formed the long running French-German tandem group. Merci, c'était génial! Dr. Beat Fasel, Dr. Constanze Pfeiffer, Corinne Eymann-Baier, Dr. Daniela Vavrecka-Sidler, Dr. Gaby Oser, and Hannah Heissler for their swift and friendly assistance in all things administration and computer hardware.

Further thanks go to all the people I met while taking little educational and inspirational breaks, detours, and distractions from the Ph.D. project: My fellow board members of the the

assistants' association of the University of Basel (avuba) for the exciting and informative time. It was great working with you to represent Ph.D. students and PostDocs at the university! Prof. Dr. Cristian Sminchisescu, and Dr. Hongyi Xu from the Machine Perception team at Google Zürich, who hosted me for four months and guided me into the world of generative human models. Diana Mincu, Dr. Jessica Schrouff, Natalie Harris, and Dr. Subhrajit Roy from the Responsible AI Health Research team at Google London, who let me join their forces for three months and provided me with a wealth of inspiration for my own research. Pritibha Singh and Nazim Ünlü for offering me the fantastic possibility to discover Novartis Data 42 part-time for 6 months; Dr. Nelly Hajizadeh for providing me with an exciting project and guidance at Data 42; and Dr. Anja Gumpinger, Giulia Capestro, Dr. Hossein Sharifi-Noghabi, Jan Dahinden, Jana Stárková, Dr. Jonathan Ziegler, Juliette Rochon, Dr. Michaela Azzarito, Dr. Sajanth Subramaniam, Dr. Silvia Zaoli, and Yesenia Cordozo-Sánchez for making me feel at home on the campus!

Last but not least, I want to thank my family and friends: My parents, Esther and Urs, and my sister Anita for their unconditional support throughout the years; my partner Dr. Oliver Müller for being an inspiring, loving, and wonderful person; my friends from school, university, dancing, and beyond for keeping my priorities straight and my work-life balance in order.

The overall project was financially supported by the Werner Siemens-Stiftung through the MIRACLE project.

Summary

Large-scale CT scans are frequently performed for forensic and diagnostic purposes, to plan and direct surgical procedures, and to track the development of bone-related diseases. This often involves radiologists who have to annotate bones manually or in a semi-automatic way, which is a time consuming task. Their annotation workload can be reduced by automated segmentation and detection of individual bones. This automation of distinct bone segmentation not only has the potential to accelerate current workflows but also opens up new possibilities for processing and presenting medical data for planning, navigation, and education.

In this thesis, we explored the use of deep learning for automating the segmentation of all individual bones within an upper-body CT scan. To do so, we had to find a network architecture that provides a good trade-off between the problem's high computational demands and the results' accuracy. After finding a baseline method and having enlarged the dataset, we set out to eliminate the most prevalent types of error. To do so, we introduced an novel method called binary-prediction-enhanced multi-class (BEM) inference, separating the task into two: Distinguishing bone from non-bone is conducted separately from identifying the individual bones. Both predictions are then merged, which leads to superior results. Another type of error is tackled by our developed architecture, the Sneaky-Net, which receives additional inputs with larger fields of view but at a smaller resolution. We can thus sneak more extensive areas of the input into the network while keeping the growth of additional pixels in check.

Overall, we present a deep-learning-based method that reliably segments most of the over one hundred distinct bones present in upper-body CT scans in an end-to-end trained matter quickly enough to be used in interactive software. Our algorithm has been included in our groups virtual reality medical image visualisation software SpectoVR with the plan to be used as one of the puzzle piece in surgical planning and navigation, as well as in the education of future doctors.

Zusammenfassung

Grossflächige CT-Scans werden häufig zu forensischen und diagnostischen Zwecken durchgeführt, um chirurgische Eingriffe zu planen und zu steuern und um die Entwicklung von Knochenerkrankungen zu verfolgen. Dabei müssen Radiolog*innen die Knochen oft manuell oder halb automatisch annotieren. Der Arbeitsaufwand dafür kann durch eine automatische Segmentierung und Erkennung einzelner Knochen verringert werden. Diese Automatisierung der Segmentierung individueller Knochen hat nicht nur das Potenzial, die derzeitigen Arbeitsabläufe zu beschleunigen, sondern eröffnet auch neue Möglichkeiten für die Verarbeitung und Darstellung medizinischer Daten für computerassistierte Chirurgie und medizinische Ausbildung.

In dieser Doktorarbeit untersuchten wir den Einsatz von Deep Learning zur Automatisierung der Segmentierung aller einzelnen Knochen in Oberkörper CT-Scans. Zu diesem Zweck mussten wir zuerst eine Netzwerkarchitektur finden, die einen guten Kompromiss zwischen den hohen Rechenanforderungen des Problems und der Genauigkeit der Ergebnisse bietet. Nachdem wir eine Basismethode gefunden und den Datensatz vergrössert hatten, machten wir uns daran, die häufigst auftretenden Fehlerarten zu eliminieren. Zu diesem Zweck haben wir die BEM-Inferenz enwickelt, welche die Aufgabe in zwei Teile aufteilt: Die Unterscheidung zwischen Knochen und Nicht-Knochen wird getrennt von der Identifizierung der einzelnen Knochen durchgeführt. Beide Vorhersagen werden dann kombiniert, was zu insgesamt besseren Resultaten führt. Eine weitere Fehlerart wird durch die Einführung des Sneaky-Nets angegangen, das zusätzliche Eingaben mit grösseren Sichtfeldern, aber geringerer Auflösung erhält. Auf diese Weise können wir größere Bereiche der Eingabe in das Netz einschleusen und gleichzeitig das Wachstum zusätzlicher Pixel in Grenzen halten.

Zusammenfassend haben wir eine auf Deep Learning basierende Methode publiziert, welche die meisten der über einhundert verschiedenen Knochen in Oberkörperscans zuverlässig segmentiert, und zwar so schnell, dass sie in interaktiver Software verwendet werden kann. Unser Algorithmus wurde in die medizinische Bildvisualisierungssoftware SpectoVR unserer Gruppe integriert, mit dem Ziel, als Puzzlestein in der chirurgischen Planung und Navigation sowie in der Ausbildung zukünftiger Ärzt*innen eingesetzt zu werden.

Chapter 1

Introduction

Computed Tomography (CT) scans are ubiquitous in medical practice. Their analysis frequently involves surgeons, or radiologists who must manually or semi-automatically annotate bones. Their annotation labour can be decreased by automating segmentation and bone detection.

We looked into using deep learning to automate the segmentation of all bones in upper-body CT scans. This automation of discrete bone segmentation has the potential to not only speed up current workflows, but also to open up new avenues for processing and displaying medical data for planning, navigation, and education.

1.1 Motivation

CT scans are three dimensional (3D) images acquired using a rotating X-ray tube. In the medical context, they have been invaluable for their ability to offer non-invasive insights. They are particularly well suited to image bone tissue. For any further processing, an accurate segmentation of bone tissue in those CT scans can prove very helpful. It can facilitate the diagnosis of bone-related diseases and the detection of bone metastases [110]. Bone segmentation can also serve as a location anchor for detecting and segmenting organs and other body structures [55]. Segmenting individual bones within a joint allows for the computation of the joint load through bone density [86]. In surgical planning and navigation applications, or radiation therapy, bone segmentations can provide semantic information and stable structural reference points [100]. However, conducting segmentations manually is a repetitive and time-consuming task. In addition, the number of medical professionals rises at a much slower pace than the medical images acquired. Therefore, automated algorithms can help fill the gap by taking over the repetitive tasks, giving doctors more time to interpret the results and talk to the patients instead of manually labelling bones [72, 105, 87].

Apart from these more general benefits of automating yet another segmentation task, we plan to use our findings in MIRACLE, our department's boldly named interdisciplinary flagship project [103]. One part of this project consists of the development of software to conduct virtual surgical planning and navigation using virtual reality (VR) and augmented reality (AR) visualisation tools [30, 145]. Currently, SpectoVR, the VR application is already used for surgical planning, and for pre-surgical patient engagement. Automatic detection of the bone in focus can help streamline the process. In addition, new advanced features for user interaction, such as quick navigation, can be designed using the bones' locations and outlines. Another current use-case of SpectoVR is to teach medical students anatomy, complementary to cadaver studies and textbooks. Adding automated segmentation of individual bones allows for the time-efficient design of educational quizzes using real-world data.

1.2 Contribution

Is the simultaneous automated segmentation of all human bones from CT possible using Deep Learning methods? This was one of the guiding questions when we started out with this project. We steered away from the well-trodden path of common segmentation problems such as brain tumours or vertebrae, where rarely more than a dozen different classes were distinguished. Instead, we plunged into a segmentation problem sporting more than 120 classes at once. Having had a minimal initial dataset at hand – 5 manually segmented CT scans from the lower and upper body each – it was a priori far from clear whether our attempts would succeed at all.

Our first publication is our proof that against all odds, the automated segmentation of 125 distinct bones of the human upper body can be achieved by supervised training and testing of a neural network on only five CT scans.

To increase the accuracy of the trained networks and the generalisability, we direly needed a larger dataset. We increased our dataset size using ensembles of our proof-of-concept models on new half-resolution data and followed up with a manual correction step. We explored the uncertainty estimation and effort estimation possible through ensemble computations.

Having established a more extensive dataset of 17 pixel-wise labelled scans and having a firm baseline to compare against, we went on to tackle more fine-grained problems: On the one hand, we realised that a large part of segmentation errors within the networks was due to mistaking background for bone, and not from mistaking one bone for another. To resolve this issue, we proposed to use a dual head segmentation network with an inference step that combines two separate predictions: A binary segmentation that exclusively separates bone tissue from the background and a prediction of the individual bone identity. We show that this leads to a decrease in this specific error type and increased segmentation accuracy. We also designed and published a synthetic dataset to further measure the capability of our method. On the other hand, we worked on the cases where bones were not identified correctly: the models seemed to confuse similar-looking bones at different body locations, such as the long bones of the arms and legs on both the left and right sides of the body. We hypothesised that this might originate from lacking global context in the model's input windows. We tackled this issue by developing a multi-resolution network that incorporates inputs of different fields-of-view and resolutions into the network.

1.3 Outline

An introduction to the medical background of bone and skeleton anatomy and to CT imaging is presented in Chapter 2. The following Chapter 3 provides technical details about deep learning.

In Chapter 4 the challenges and peculiarities of distinct bone segmentation are discussed, and an overview of prior work in the area is given.

Chapters 5 to 8 contain the four publications and technical reports forming this thesis's main part. In Chapter 5 we present a baseline solution for automated distinct bone segmentation on our small initial dataset. Chapter 6 details our steps of increasing the dataset size while analysing the use of ensembles to improve results and minimise annotator time. We present a solution to commonly encountered labelling mistakes between background and foreground in Chapter 7. Chapter 8 contains a way of providing an increased field-of-view to the segmentation networks while keeping the computational burden small. Conclusions and an outlook are finally presented in Chapter 9.

Chapter 2

Medical Background

This chapter starts in Section 2.1 with an overview of the anatomical and physiological features of bones. The human skeleton's anatomy and its modes of variation are explained in Section 2.2. We then cover CT, our image modality of interest, in Section 2.3. Some of the envisioned use cases for our algorithm, such as surgical planning and anatomical education, are introduced in Section 2.4.

2.1 Bone

Bones are rigid organs of complex structure that form part of the skeleton in humans and most vertebrate animals. Bones come in various shapes and compositions, optimised for their wide range of functions [84]. Those functions include the formation of blood cells, mechanical support, protection, and mineral homeostasis [19].

2.1.1 Bone structure and tissues

Bone consists of two main layers, cortical bone and cancellous bone, as illustrated in Figure 2.1. The dense cortical bone, which carries the body's weight, makes up the majority of the bone mass. The softer cancellous bone, sometimes called the spongy or trabecular bone, diverts pressures to the cortical bone and thus helps bear the load. On the outside, every bone is enclosed in the periosteum, a dense layer of fibrous tissue that enhances the resistance to mechanical stress [95]. Located on the inside of the cancellous bone is bone marrow, a semi-solid tissue that stores fat (yellow marrow) and is the predominant site of blood cell formation (red marrow) [18].

In total, bone tissue contains around 10% water, 25% organic material, and 65% minerals when calculated based on weight [18]. The organic material is mostly type I collagen, which provides tensile strength and elasticity in the bone structure. The minerals, on the other hand, give the material its rigidity.

The basic structural and functional units of compact bone are called osteons. They are cylindrical and contain concentric layers of compact bone tissue. A canal with blood supply is located at the centre of the osteon. An illustration of the structure can be found in Figure 2.1

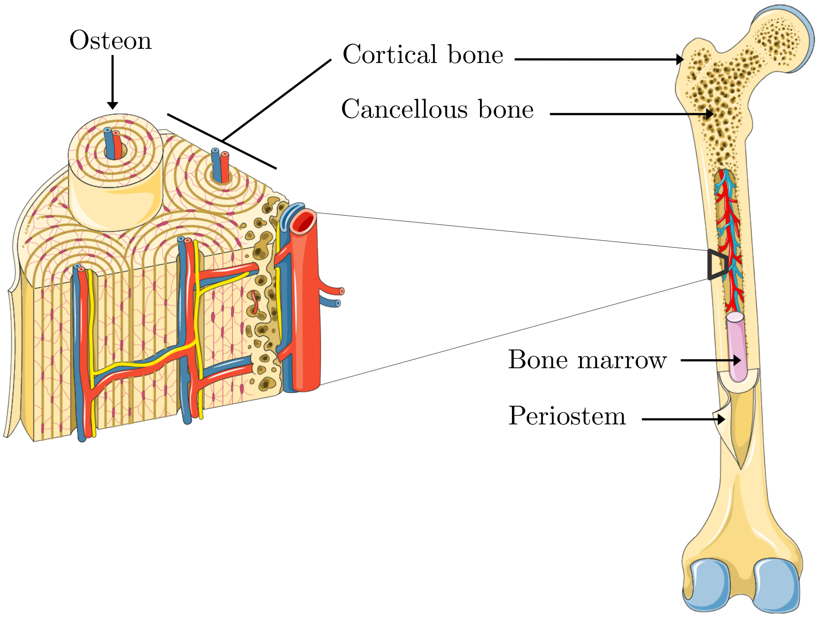


Figure 2.1: Left: Schematic detail of the cortical bone. Highlighted are the osteons, the cortical bone's primary functional units with a cylindrical shape. Right: Schematic of a femur, displaying the main tissues.

Figure modified with lines and text annotation after adaptation of "bone structure" and "osteon" from Servier Medical Art by Servier, licensed under a Creative Commons Attribution 3.0 Unported License.

Osteons vary in size but are around $0.3\,\mathrm{mm}$ in diameter and $1\,\mathrm{mm}$ long [114]. They usually run parallel to a bone's long axis.

2.1.2 Bone shape classification

Bones can be grouped according to their shape [95, 114]. Figure 2.2 complements the bone groups mentioned in the following list:

- **Long bones** are present in the arms and legs and consist of a shaft (diaphysis) and two epiphyses at the ends.
- **Short bones** such as the carpal and tarsal bones are found in the wrists and parts of the feet.
- **Flat bones** are made up of two layers of cortical bone surrounding cancellous bone. Flat bones are thin and, contrary to their name, tend to be curved. Flat bones include the skull vault and the ribs.
- **Irregular bones** Many of the bones of the face, skull, and vertebrae do not fit the previous categories and are therefore classified as irregular.

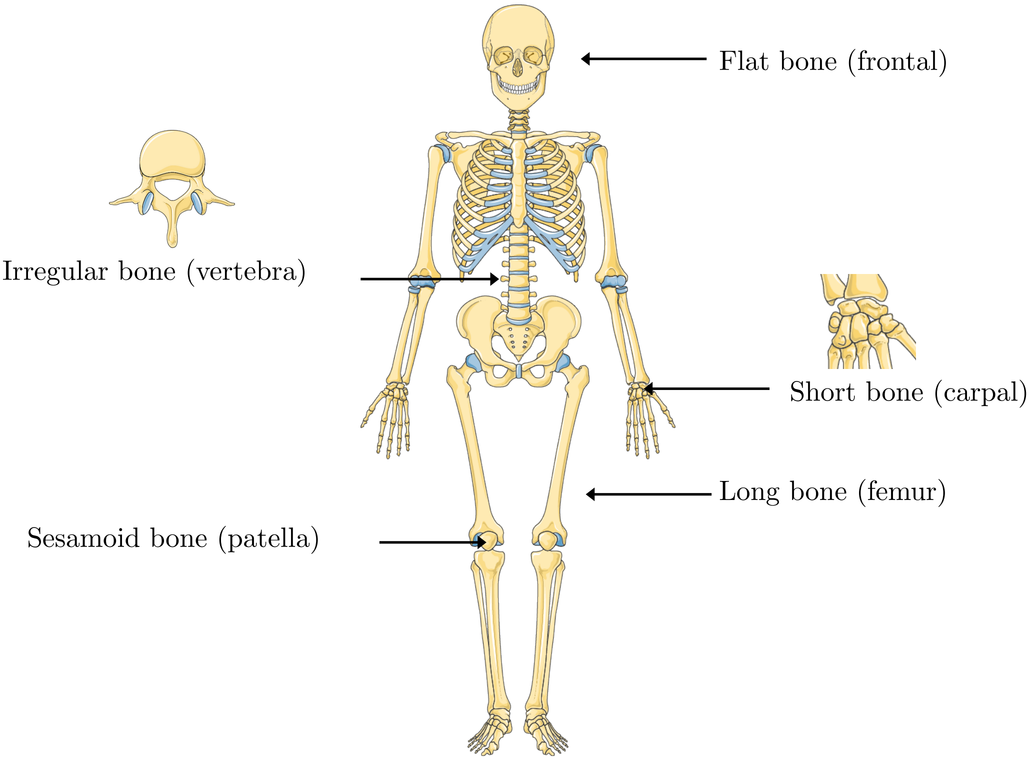


Figure 2.2: Left: Schematic of the adult human skeleton with examples for different bone shape groups.

Figure modified with lines and text annotation after adaptation of ”skeleton”, ”arm”, and ”vertebra” from Servier Medical Art by Servier, licensed under a Creative Commons Attribution 3.0 Unported License.

- **Sesamoid bones** are bones that are embedded in tendons. The biggest sesamoid bone in humans is the patella which is embedded in the tendons of the knee.

- **Accessory bones** are anomalous supernumerary bones that typically originate from a failure of fusion of the ossification centres.

2.2 Skeletal Anatomy

The adult human skeleton is characterised by an arrangement of bones and joints that link them. On average, it consists of 206 bones that account for 14 percent of the total body weight [125]. Occasionally, extra accessory bones are present and increase the number, such as an additional lumbar [125].

Conventionally, the skeleton is divided into two groups: (i) The axial skeleton, which is composed of the torso and cranial bones. It maintains the human’s upright posture and transmits the weight to the lower extremities. (ii) The appendicular skeleton, which consists of the bones of the limbs and enables movement [125]. A list of these bones and how they were treated in our segmentation tasks is presented in Table 2.1 and Table 2.2.

Bones show considerable variations from one individual to another, which adds to the challenge of bone segmentation. Variations in the skeleton are mostly due to the following four factors [137]:

- **Age**: Bones develop from more than 800 centres of ossification, which eventually coalesce, resulting in a considerable variation in the number of bones in children [125]. Not only the number but also the size and shape of bones continue to change during childhood [137].
- **Sex**: There is a slight sexual dimorphism in human bone size, albeit with considerable overlap in the distributions. Some elements of the skull and of the pelvis tend to show variations between the sexes, such as the subpubic concavity [137].
- **Geography/Population**: There are no human skeletal markers that exactly correspond to geographic origin, but there are local tendencies of variation such as in nasal bones or cheekbones [137].
- **Individual**: The skeletons of individuals of the same age, sex, and geographic origin can differ substantially, which is referred to as individual, or idiosyncratic variation [137]. Changes in the number of bones observed can occur through supernumerary bones, such as an additional lumbar vertebra, or pairs of ribs located at the cervical or lumber vertebrae.

In conclusion, while the number of bones and their relative position in the human skeleton is quite similar in human adults, there is still considerable variation in the shape and size of bones among individuals.

Table 2.1: Bones of the axial skeleton of human adults [125]. In the third column, we indicate the bones labelled for our distinct bone segmentation tasks. Simplifications in the head area have led to the distinction of only two labels, one for the skull and one for the mandible.

Area	Anatomy	Data labels
Braincase	ethmoid	skull
(8 bones)	frontal	
	occipital	
	parietals (×2)	
	sphenoid	
	temporals (×2)	
Face	conchae (×2)	skull
(14 bones)	lacrimals (×2)	
	maxillae (×2)	
	nasals (×2)	
	palatines (×2)	
	vomer	
	zygomatics (×2)	
	mandible	mandible
Ear	incus (×2)	skull
(6 bones)	malleus (×2)	
	stapes	
Throat	hyoid	hyoid
(1 bone)		
Vertebral column	cervical vertebrae (×7)	cervical vertebrae (×7)
(26 bones)	thoracic vertebrae (×12)	thoracic vertebrae (×12)
	lumbar vertebrae (×5)	lumbar vertebrae (×5)
	sacrum	sacrum
	coccyx	
Thorax	ribs (×24)	ribs (×24)
(25 bones)	sternum	sternum
Pectoral girdle	clavicles (×2)	ribs clavicles (×2)
(4 bones)	scapulae (×2)	scapulae (×2)

Table 2.2: Bones of the appendicular skeleton of human adults [125]. We indicate the bones labelled for our distinct bone segmentation tasks in the third column. Unlike the axial skeleton, no simplifications have been made, but the sesamoids of the thumb have been added.

Area	Anatomy	Data labels
Arm (6 bones)	humeri (×2) radii (×2) ulnae (×2)	humeri (×2) radii (×2) ulnae (×2)
Hand (54 bones)	capitates (×2) hamates (×2) lunates (×2) pisiforms (×2) scaphoids (×2) trapeziums (×2) trapezoids (×2) triquetrals (×2) metacarpals (×10) phalanges (×28)	capitates (×2) hamates (×2) lunates (×2) pisiforms (×2) scaphoids (×2) trapeziums (×2) trapezoids (×2) triquetrals (×2) metacarpals (×10) phalanges (×28) **sesamoids** (×2)
Pelvic girdle (2 bones)	coxae (×2)	coxae (×2)
Leg (8 bones)	femora (×2) fibulae (×2) patellae (×2) tibiae (×2)	femora (×2) fibulae (×2) patellae (×2) tibiae (×2)
Foot (52 bones)	calcanea (×2) cuboids (×2) intermediate cuneiforms (×2) lateral cuneiforms (×2) medial cuneiforms (×2) naviculars (×2) tali (×2) metatarsals (×10) phalanges (×28)	calcanea (×2) cuboids (×2) intermediate cuneiforms (×2) lateral cuneiforms (×2) medial cuneiforms (×2) naviculars (×2) tali (×2) metacarpals (×10) phalanges (×28)

2.3 Computed Tomography

Computed Tomography is an X-ray-based imaging technique that permits cut-free insights into the interior of human bodies in three dimensions.

In clinical practice, CT scans are extensively used. For instance, in Switzerland in the year 2020 alone, 344 CT devices [90] were used to conduct over one million exams [91], for a population of 8.6 million. This makes CT examinations still more prevalent than the radiation-free MRI [92, 93].

CT images are taken by probing the human body, or any other object of interest, with X-ray beams. To varying degrees, X-rays are absorbed as they pass through the different tissues of the body. A detector array measures the beam's intensity after it leaves the body. To allow the reconstruction of three-dimensional structures, the body must be irradiated from many directions. Commonly, this is accomplished by a measuring method in which the X-ray source moves in a spiralling motion while the examination table moves linearly along the rotating axis, resulting in a continuos spiral scan path [20]. The three-dimensional image then needs to be reconstructed using the recorded projections, which results in a multidimensional inverse problem.

After reconstruction, the obtained attenuation values μ are standardised by scaling them relative to the attenuation coefficient of water μ_w. This process yields values in the so-called Hounsfield Unit (HU), named for one of the principle's inventors [19].

$$HU = 1000\frac{\mu - \mu_w}{\mu_w} \tag{2.1}$$

In contrast to MRI scans, that vary greatly in contrast according to the sequence and the vendor, HU values in CT images are fairly consistent even when acquired on different scanners. Typical HU values are -1'000 for air and 0 for water. Soft tissue and organs largely consist of water and have a HU value close to 0. Bone, which contains only about 10% water but large quantities of X-ray absorbing minerals, has higher HU values. Cortical bone has the highest value among body tissues, ranging from several hundred to 1'000+ HU. Cancellous bone tissue has lower HU values that are already in the range of some soft tissues [20]. This similarity in HU values impedes a precise identification of bone using a HU threshold alone. The most extreme values are encountered in metal. Metal, such as found in implants, can theoretically exceed 10'000 HU. However, because CT images are usually stored as 12-bit data, values are truncated to stay in the range from -1'024 to 3'071 [40]. Nevertheless, implants are still a common source of imaging artefacts.

CT scans are susceptible to several possible artefacts that degrade image quality. The following are of importance to our work:

- **Streak artefacts:** They are typically found near X-ray-blocking materials. Common examples are metal implants [148]. In the work for this thesis, we eliminated CT images of patients with joint implants from our dataset and predominantly observed this distortion around the mandible due to dental fillings. An example is shown in Figure 2.3.

- **Partial volume effect:** This artefact is created mainly through the resolution of the image, in which multiple tissue types appear within a single voxel. The results for a small amount of high-density area such as bone leads to the same intensity value as a larger area of soft-tissue [10]. In Figure 2.3 the partial volume effect can be observed in the cancellous bone of the skull.

- **Noise:** A poor signal-to-noise ratio has a detrimental impact on an image. When applying low doses of radiation or when the slices are particularly thin, grain-like noise develops more frequently.

In general, the presence of imaging artefacts makes downstream tasks harder. Consequently, much research has been conducted to reduce all kinds of artefacts and increase the imaging quality while keeping the radiation doses low. While elevated radiation doses are required for higher resolution CT images [23], they lead to undesirable effects, most notably an increase in cancer risk. Therefore, the goal is to reduce radiation exposure to a minimum.

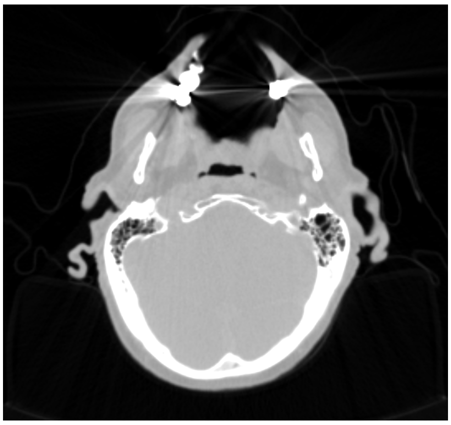


Figure 2.3: Axial slice of a CT, displaying common imaging artefacts. Strike artifices are visible in the centre, originating from tooth fillings. The partial volume effect can be observed in the cancellous bone of the skull, where the bone structures are thinner than the image resolution.

2.3.1 Visualisation

An accurate and understandable depiction of medical images is essential for diagnosis and surgical planning.

In CT scans, the intensity range of the images can be rather large. Very high or very low numbers may dominate the image when displaying or printing the scan, making it challenging to differentiate intermediate values at a suitable resolution. This issue is resolved by a method known as windowing. In windowing, a particular range of values is mapped onto the greyscale gradient. For this to work, a maximum and minimum threshold must be established. Any values below the threshold will appear in black, while any numbers above will be displayed in white. The threshold values are determined based on the application and usually manually selected by radiologists.

CT images are naturally three-dimensional, making intuitive viewing on a two-dimensional computer screen challenging. Displaying CT scans one slice at a time and requiring the viewer

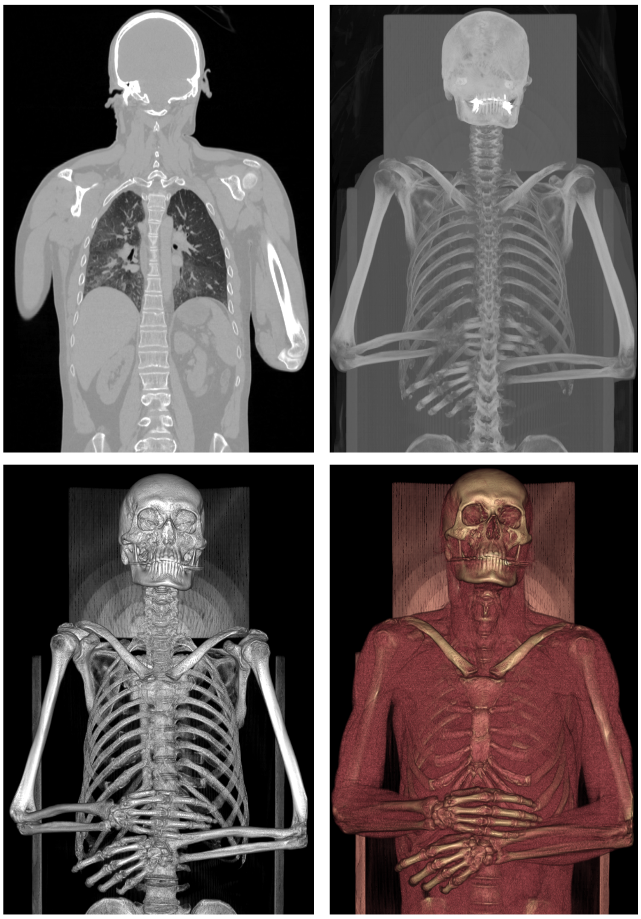


Figure 2.4: Different CT visualisation techniques. Top left: a single frontal slice of a CT scan. Top right: A maximum intensity projection. Bottom: Two examples of volume rendering with different transfer functions.

to go through each two-dimensional slice may be the simplest way. However, it requires the viewer to mentally perform the 3D-mapping task while constantly interacting with the software [35].

Using more advanced displaying methods can enhance the viewing experience. Maximum intensity projections and average intensity projections use information from all or multiple slices simultaneously and provide an image similar to an X-ray; see also Figure 2.4 top right.

Direct volume rendering techniques, such as ray casting [28, 4], are more complex. They are still two-dimensional projections, but they allow for three-dimensional-looking results. See Figure 2.4, bottom row, for an example of ray casting using two different transfer functions. Ray casting works on physical principles of emission, absorption, and scattering and simulates the way of a ray of sight through a volume into a two-dimensional image. A simplified graphical explanation of the idea is provided in Figure 2.5. As a first step, a ray is cast from the eye-point through the image and through the volume. For every pixel of the final image, one ray is needed. Then, along each of these rays, equidistant points are sampled inside the volume. These sampling points can generally be positioned between voxels, which necessitates interpolation. Colour, illumination and transparency values are determined using a transfer function for each sampling point. For example, a simple transfer function to highlight bones in CT would map the colour white and full opacity to HU values in the range of bones, and full transparency everywhere else. Finally, the sampling point values along a ray are merged into a single colour value for the current pixel, similar to applying a stack of foils on an overhead projector [135].

To speed up the process, most implementations use early ray termination. The sampling points are then only evaluated up to the point where the opacity channel is almost saturated. Anything behind this point will be hidden behind opaque materials, and further computations would not contribute to the final result [69].

Whilst volume rendering with a transfer function enables the visualisation of a tissue of choice, it cannot differentiate between instances of the tissue, such as identifying distinct bones.

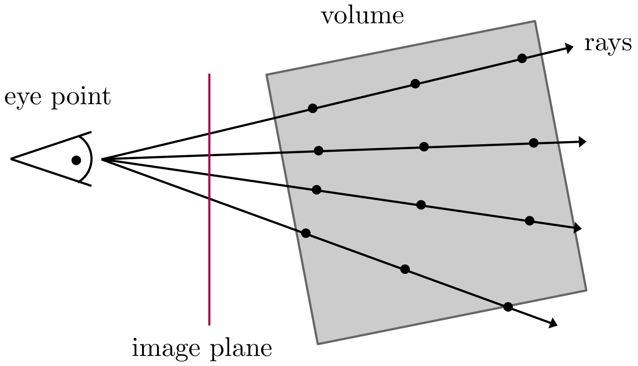


Figure 2.5: Raycasting idea: For every final image pixel, one ray is cast. The rays start at the eye point, traverse the image plan, and finally, the volume. Sample points are generated along the rays inside the volume. The optical properties at this point are then evaluated as a function of the data value. In a final step, the optical properties are accumulated along each ray.

2.4 Surgical Planning and Anatomical Education in virtual reality

For more than 25 years, volume rendering has been performed in in medical applications [53]. Only more recently have virtual reality (VR) headsets come to a stage of development where they are user-friendly and cost-efficient enough to be considered for routine clinical use. Specifically, their frame rate first had to be significantly increased so that users would not get nauseous while using the devices. However, current technology enables the real-time volume rendering of medical data in VR in an efficient manner [30]. Using VR in combination with raycasting allows for an immersive experience and for quick and intuitive changes in the viewing direction. This renders details visible that might be hidden when only viewing the two-dimensional projections. Furthermore, the use of VR in surgical planning has been shown to be beneficial in terms of surgical outcome [134, 118].

Anatomical knowledge is a fundamental ability taught to medical students throughout their first year of study. In teaching anatomy, cadavers are the gold standard. They enable the study of real human bodies from all angles, including minute details, and provide force and touch feedback. Cadavers also display a broad spectrum of human physiology, usually lacking in textbooks. Due to limits in supervision, expenditures, and ethical considerations, the window of opportunity for students to research with these specimens is quite limited [85]. Therefore, AR and VR technologies can let students benefit from immersive experiences, and complement cadaver dissection classrooms.

Some studies show that the use of AR and VR indeed improves anatomy teaching. In [124] for example, the use of a VR environment to visualise magnetic resonance cholangiopancreatography (MRCP) was shown to improve the understanding of biliary anatomy and intraoperative performance among surgical trainees, compared to trainees who only prepared using conventional visualisation of MRCP.

In [32], the use of augmented reality (AR) has been demonstrated to improve the test performance of students who were required to learn the anatomy of the foot muscles, compared to students who learned using only notes, photographs, and videos. A comparable study found that students who used AR to study human gross anatomy were more motivated and had a much better comprehension of the three dimensional structure compared to those who used standard textbooks [79, 64].

We aim to combine volume rendering in VR with automated segmentation to develop new and improved workflows in surgical planning and in teaching anatomy. A screenshot of our current segmentation and volume rendering combination is shown in Figure 2.6.

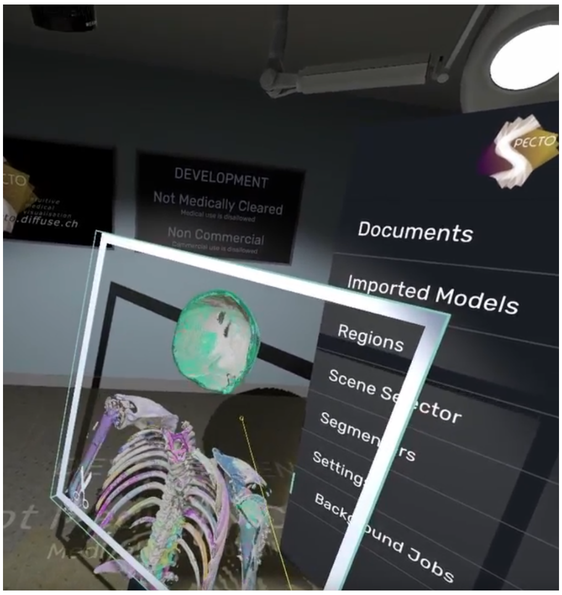


Figure 2.6: Screenshot of the SpectoVR software, taken by Norbert Zentai from our planning and navigation group. The SpectoVR software uses volume rendering and transfer functions to display the bone tissue only. In conjunction with the distinct bone segmentation algorithm developed in this thesis, distinct bones can be distinguished (indicated by different colours).

Chapter 3

Deep Learning

In this chapter, we provide an overview of deep learning. We start by explaining the commonly used components of neural networks in Section 3.1 and introduce their training procedure in Section 3.2.

3.1 Artificial Neural Networks

In recent years, machine learning and specifically deep learning has become a ubiquitous tool in various study disciplines due to spectacular advances in natural language processing, computer vision, and control learning. Figure 3.1 illustrates the relationship between several of the terms frequently encountered in the context of deep learning.

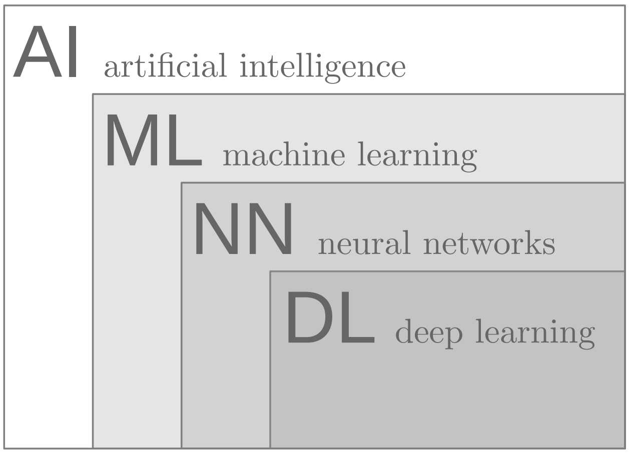


Figure 3.1: Relations of terms used around deep learning.

Alan Turing published foundational theoretical work on artificial intelligence as early as 1950 [131]. Shortly after that, in 1956, the so-called Dartmouth Conference was held, credited as the actual birthplace of artificial intelligence as a scientific discipline [61]. The perceptron

[108], a one-layer artificial neural network (ANN) intended for image recognition, produced the first significant practical results another two years later, in 1958. In the subsequent decades, many machine learning algorithms have been developed that either use handmade features, such as the support vector machine (SVM), or learn a data representation on the fly, such as neural networks.

Most of today's methods in computer vision and hence medical image analysis and processing centre around deep learning. In deep learning, artificial neural networks with many layers are trained on potentially massive datasets to fulfil a plethora of tasks.

3.1.1 Fully connected networks

Artificial neural network (ANN)s are machine learning models that were originally inspired by biological neural networks, as found in the brain. Its most basic and original form, the perceptron, is a binary linear classifier used to learn a threshold function:

$$f(x) = y = \begin{cases} 1, & \text{if } \sum_{i=1}^{N} w_i x_i + b > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

We denote the trainable parameters, also commonly called weights, as $w = (w_1, ..., w_N)$ and the inputs as $x = (x_1, .., x_N)$. Examples of possible inputs might be the pixel intensities of an image. The bias b allows for a shift of the decision boundary away from the origin.

A perceptron can have an arbitrary number of inputs and thus number of weights, but it will always result in a linear decision function. As a matter of fact, it is incapable of modelling arbitrary functions such as XOR. This issue, which was extensively mentioned by [82], led to a crisis in the artificial intelligence research field in the 1970ies, the so-called AI winter.

One of the reasons ANNs experienced a revival ten years later is the broad realisation that the restrictions mentioned above are practically irrelevant. Adding more layers to the ANN makes it possible to overcome the limitations of linear functions. In fact, it can be demonstrated that multilayer ANNs with one hidden layer of adequate width can already serve as universal approximators for a wide variety of useful function classes, such as the solutions to high-dimensional optimisation problems [46].

These additional layers introduced between the input and output are typically referred to as hidden layers. Adding more or wider hidden layers increases not just the mathematical expressiveness but also the computational complexity of the network. An input of size N and a hidden layer with M nodes require the concurrent management of $N \cdot M$ network weights. Figure 3.2 depicts a schematic illustration of an ANN that acts on the pixels of a 4×4 image ($N = 4 \cdot 4 = 16$) and has a first hidden layer with a width of $M = 2$ nodes. For big M and N, and even more so if there are a large number of subsequent hidden layers, this quickly results in an enormous number of weights. Efficient and parallel computation of so many values is far from trivial. Consequently, these computational requirements have long hampered the practical use of large and deep networks until the emergence of GPUs for tensor computations has led to vast improvements.

Another problem emerges not with the addition of more layers but as a result of the fully connected structure of the network: Each node in the hidden layer receives weights from every

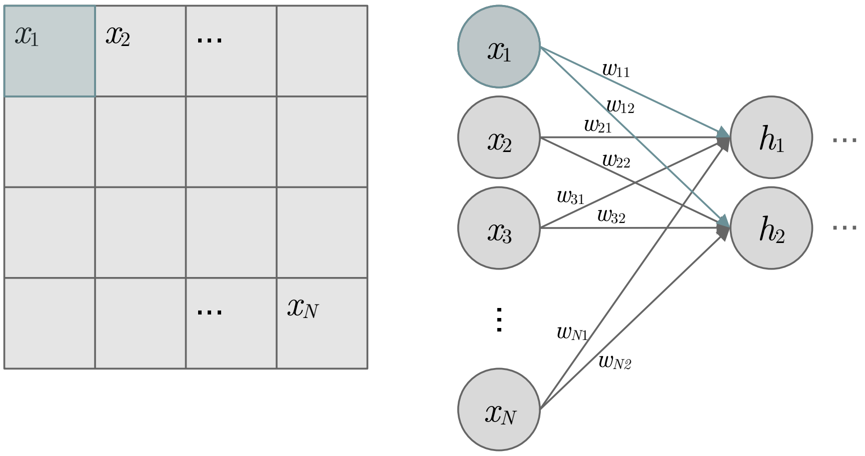


Figure 3.2: Pixels of a 4×4 image (left) as input for an ANN with a first hidden layer of width $M = 2$ (right).

node in the layer preceding it. This means that when working with an image as input, all spatial information within the image is lost. The relationship between inputs x_1 and x_2 from adjacent pixels is treated identically to the relationship between x_1 and x_N.

Fully connected networks also have other shortcomings: The way the network is set up, weights are tightly coupled to input pixels. This leads to very poor generalisation ability to recognise input patterns that are slightly translated [67]. It would also not be possible to use the network on an input image of smaller or larger size than the images in the original training set.

3.1.2 Convolutions

Many of the problems that occur with fully connected networks can be alleviated by using convolutional neural networks (CNNs). CNNs are a special kind of ANNs that use convolution operations in one or more layers [41]. They are very well suited for processing data with a grid-like topology, such as 1D time-series data, 2D natural images, 3D CT images, or 4D flow MRI.

A convolution is a mathematical operation on two functions f and g that results in another function $(f * g)$. It is defined as

$$c(t) = (f * g)\,(t) := \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau. \tag{3.2}$$

In the context of ANNs, we usually convolve a finite discrete input I and a finite discrete kernel K. The kernel is applied to subsequent input regions to produce an output tensor C, as illustrated in Figure 3.3. In the three-dimensional case of a CT image as input, the result of the convolution is given by

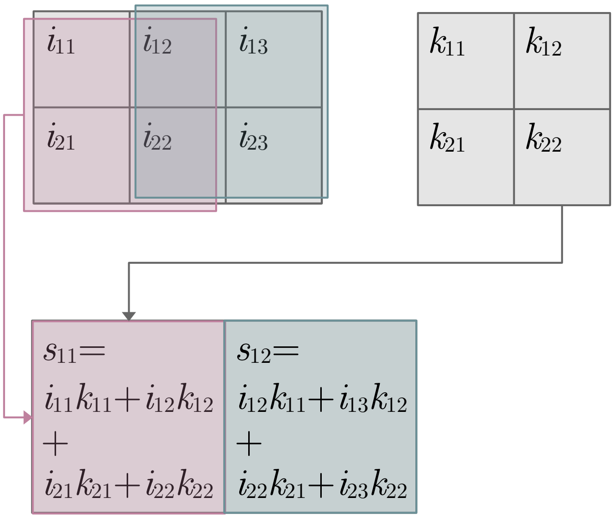

i_{11}
i_{12}
i_{13}
i_{21}
i_{22}
i_{23}
k_{11}
k_{12}
k_{21}
k_{22}
$s_{11} = i_{11}k_{11} + i_{12}k_{12} + i_{21}k_{21} + i_{22}k_{22}$
$s_{12} = i_{12}k_{11} + i_{13}k_{12} + i_{22}k_{21} + i_{23}k_{22}$

Figure 3.3: An example of a 2D convolution shown on an input image I of size 2×3 and a kernel K of size 2×2. We restrict the application of the kernel to input regions that lie fully within I and therefore get an output S of size 1×2, which is smaller than the input.

$$C(I) = (K * I)(x, y, z) = \sum_{k_x} \sum_{x_y} \sum_{k_z} I(x - k_x, y - k_y, z - k_z) K(k_x, k_y, k_z). \tag{3.3}$$

Typically, kernels are much smaller than the inputs, with $3 \times 3 \times 3$ and $5 \times 5 \times 5$ being popular choices for 3D networks. Furthermore, there is usually not just one kernel but many. Each kernel is applied to the same input and produces its own output tensor. These outputs of individual kernels are called channels.

In classical image processing, handcrafted kernels have been used to extract edges and other features from images. Sobel filters are a particularly popular choice. They are discrete differentiation operators that compute an approximation of the image gradient at every location in both horizontal and vertical directions. As a result, they highlight the areas where intensities change and thus detect edges in the image. In contrast to Sobel filters and comparable handcrafted kernels, the convolutional kernels of a CNN are iteratively updated during the optimisation process. For some examples of both Sobel filters and trained kernels, see Figure 3.4.

CNNs have many favourable properties [41]:

- **Sparse interactions:** By using kernels much smaller than the input, only spatially close variables of the input interact with one another during the computation of the intermediate output. In a fully connected network, there are $\mathcal{O}(M \cdot N)$ computations necessary to arrive from an input layer of size M to an output layer of size N. Using convolutions with a kernel of size L, this number drops to $\mathcal{O}(M \cdot L)$ with $L << N$.

- **Parameter sharing:** Because the same kernel is applied to all input sections, the memory requirement to store trainable weights decreases dramatically. In the fully connected case, the storage of $(M \cdot N)$ weights is required. Using convolutions, only L weights need to be stored, which decouples the input size from the number of trainable parameters.

- **Equivariant representations:** Thanks to its parameter sharing property, convolutional layers are equivariant to a translation t, $C(t(I)) = t(C(I))$. In other words, the outcome is identical if we first translate our input image and then perform convolution or if we apply translation after convolution.

These properties make CNNs ideal candidates to work with image data.

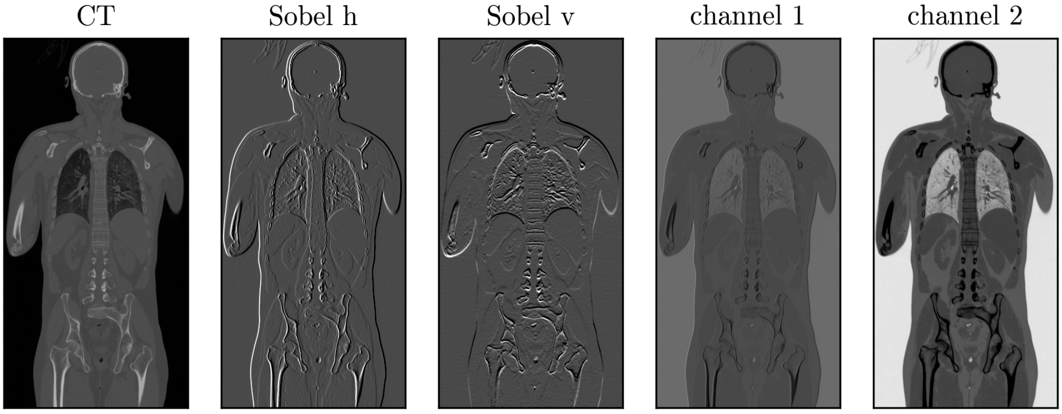


Figure 3.4: From left to right: A frontal slice of a CT scan of our dataset. Two examples of the same scan convolved with a horizontal and a vertical Sobel filter. Two examples of the same scan convolved with kernels taken from one of our trained models.

3.1.3 Downpooling, strided convolutions, transposed convolutions

Pooling layers are conceptually very similar to convolutional layers, with the difference that their kernels are fixed, and no parameters have to be learned. Usual choices are the max-pool and the average-pool kernels, which map the maximum or average of their kernel region to one output value. The application of such pooling kernels enforces an approximate invariance to small translations of the input [41]. However, downsampling is the main application of pooling in CNNs. By choosing pooling regions that are s pixels apart instead of 1 pixel, the output's spatial size is roughly reduced by a factor of s per dimension. In this case, the term downpooling, or strided pooling with a stride of s is generally used. The same principle can also be applied to convolutions, which are then called strided convolutions and reduce the output layer's spatial size. Using the same stride in all dimensions, Equation (3.3) turns into:

$$C_s(I) = (K * I)\,(x, y, z, s) = \sum_{k_x} \sum_{x_y} \sum_{k_z} I\,(sx - k_x, sy - k_y, sz - k_z)\, K\,(k_x, k_y, k_z)\,. \tag{3.4}$$

The strided convolution Equation (3.4) is a generalisation of Equation (3.3) which used a unit-stride of $s = 1$. Strided convolutions or poolings achieve the same effect as their unit-strided $s = 1$ version followed by a separate downsampling step, but require fewer computations, see also Figure 3.5.

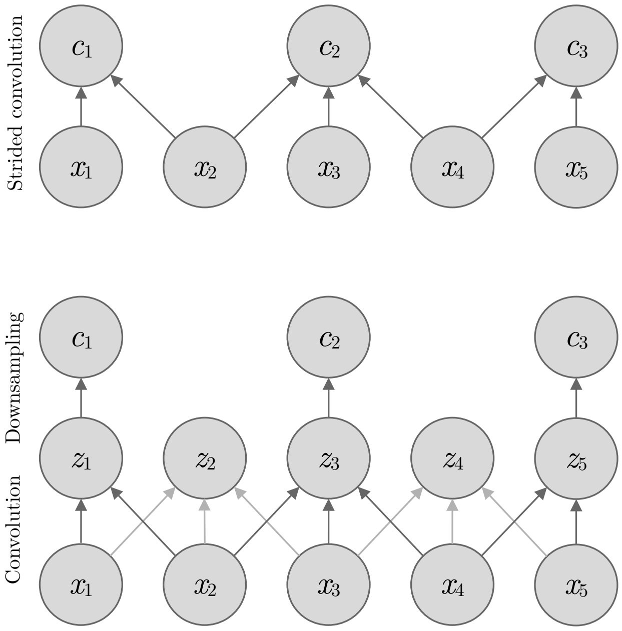


Figure 3.5: A strided convolution with stride $s = 2$ produces the same output as a convolution followed by a separate downpooling step of stride 2. Top: 1D convolution with stride $s = 2$ and a kernel of size 3. Bottom: 1D convolution with unit stride $(s = 1)$ followed by a downsampling step of stride 2. Both approaches are mathematically equivalent, but the two-stage approach below performs computations that are never used (light grey arrows).

Transposed convolutions serve the opposite purpose of strided convolutions [27]. They produce an output of higher spatial resolution than their input. Alternatively, classical upsampling methods without trainable parameters, such as linear interpolation, can be used to increase the resolution of the output.

3.1.4 Activation functions

Activation functions are used after convolutional or fully connected layers to increase the possible complexity of the function being learned. In particular, activation functions such as softmax are also used at the very end of a network. Equation (3.1) can be rewritten as

$$a(\mathbf{Wx} + b) \tag{3.5}$$

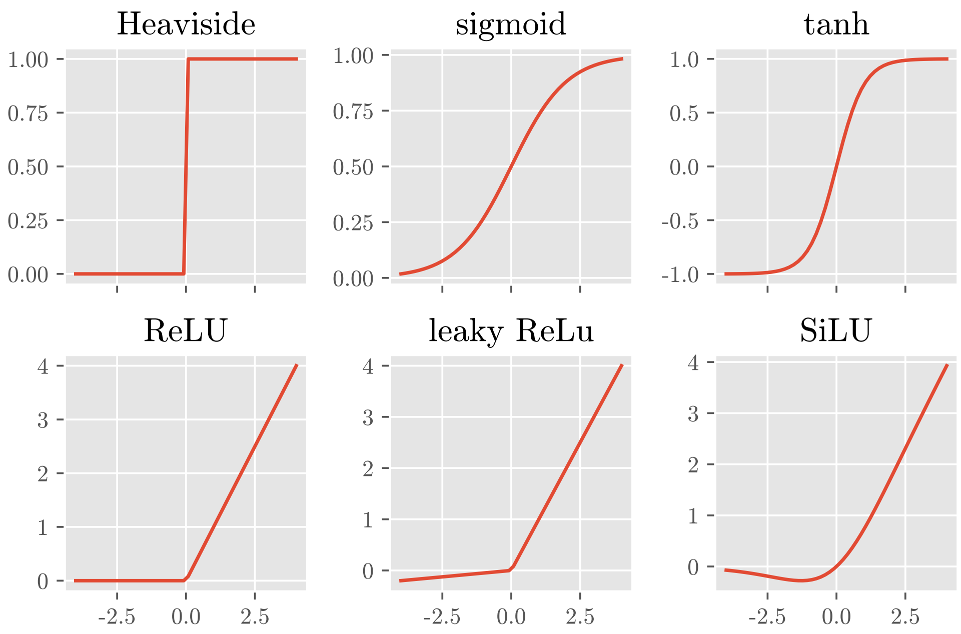


Figure 3.6: Selection of activation functions used within ANNs.

using the Heaviside step function

$$H(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \tag{3.6}$$

as activation function a. A graphical representation of this and the following activation functions can be found in Figure 3.6

The sigmoid function is widely used as a continuous replacement for the Heaviside function to aid training convergence:

$$\sigma(x) = \frac{1}{1 + \mathrm{e}^{-x}} \tag{3.7}$$

Hyperbolic tangent (tanh) is an activation function similar in shape to the sigmoid but projecting the values on a continuous curve between -1 and 1 instead of 0 and 1.

$$\tanh(x) = \frac{\mathrm{e}^{x} - \mathrm{e}^{-x}}{\mathrm{e}^{x} + \mathrm{e}^{-x}} \tag{3.8}$$

To avoid training problems caused by vanishing gradients, the use of rectified linear units (ReLUs)

$$\begin{aligned} \mathrm{ReLU}(x) &= \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} \\ &= \max(0, x) \end{aligned} \tag{3.9}$$

and its variants such as leaky ReLU as activation function is wide-spread [120].

$$\mathrm{leReLU}(x) = \begin{cases} \alpha x & \text{if } x < 0,\ \alpha \approx 0.01 \\ x & \text{if } x \geq 0 \end{cases} \tag{3.10}$$

A smoother variant is known as Sigmoid linear unit (SiLU) [3].

$$\mathrm{SiLu}(x) = \frac{x}{1 + \mathrm{e}^{-x}} \tag{3.11}$$

3.1.5 Normalisation

Normalisation layers have proven to be highly beneficial in facilitating the learning of vast parameter spaces. Normalisation can be performed along multiple dimensions. Commonly, parameters are normalised along the batch dimension:

$$BN(y) = \gamma \frac{y - \mu}{\sigma} + \beta\,, \tag{3.12}$$

where μ is the mean, and σ is the variance of the ys from the different batches.

Normalisation has multiple advantages: it decreases training time, reduces covariate shift, and has a regularising effect, resulting in improved generalisation quality of the model [48]. While batch-normalisation is the de facto standard for the majority of use-cases, the situation is different for the many 3D neural network applications. In the case of 3D networks, other types of normalisations, such as instance [132], layer [8], and group [141] normalisation, are more prevalent since limited computational memory necessitates working with very tiny batch sizes.

3.1.6 Classification networks

Straight-forward classification networks are built using a series of convolutions and down-pooling layers, followed by one or multiple fully connected layers, eventually ending up with as many output nodes as classes (N_{classes}) [66, 63, 121].

The input may contain information on many scales. Convolutions, however, only allow for interactions of neighbourhoods of pixels that are the same size as the convolutional kernel. This neighbourhood of interacting inputs around a single unit is called the unit's receptive field. A graphical explanation of the receptive field is given in Figure 3.7. By applying subsequent convolutional layers, a network's receptive field gradually increases. The receptive field of a unit in a CNN of L layers with only one path can be calculated using the following recursive formula [6]:

$$r_L = 1 \tag{3.13}$$

$$r_{\ell-1} = s_\ell r_\ell + (k_\ell - s_\ell) \tag{3.14}$$

$$r_0 = \sum_{\ell=1}^{L} \left((k_\ell - 1) \prod_{i=1}^{\ell-1} s_i \right) + 1 \tag{3.15}$$

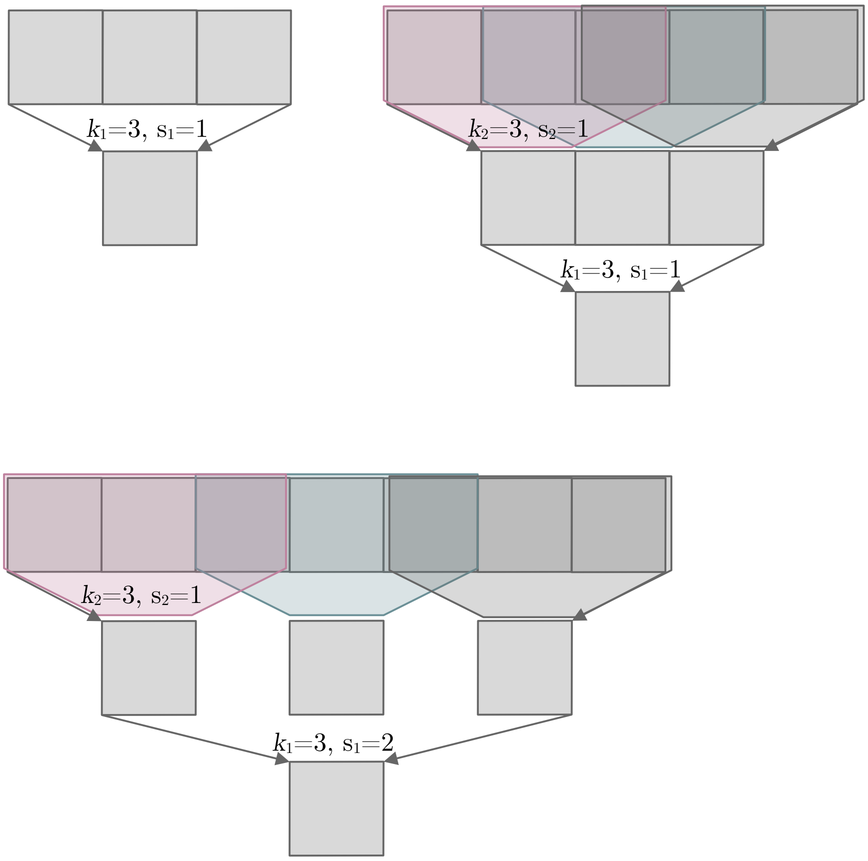


Figure 3.7: Receptive fields illustrated using 1D convolutions. Top left: a single convolution with kernel size 3 leads to a receptive field of size 3. Top right: two subsequent convolutions of size 3 lead to a receptive field of size 5. Bottom: A unit-stride convolution of size 3 is followed by a stride-2 convolution or down-pooling, which leads to a receptive field of size 7.

Here r_ℓ denotes the receptive field size at layer $0 \leq \ell \leq L$. The kernel size and stride at layer ℓ are denoted using k_ℓ and s_ℓ. Using a series of convolutions and strided convolutions (or pooling layers) steadily increases the receptive field while reducing the actual spatial resolution of the output. This compression in the spatial dimension is usually offset by the inclusion of multiple convolutional kernels per layer, which increases the number of output channels.

3.1.7 Segmentation networks

When removing the fully connected layers from the classification network above, the output has a small spatial size but a large number of channels. This is usually referred to as a latent encoding of the input. To obtain a pixel-wise classification, i.e. a segmentation of the input, upsampling steps are necessary [77, 107]. This happens by adding subsequent up-pooling or transposed convolution layers until the size of the network output matches the segmentation target size. One very successful example of segmentation networks, the U-Net [107], does this with symmetric contraction and expansion parts of the network, as illustrated in Figure 3.8. To aid optimisation convergence and to recover spatial information that might get lost during contraction, skip connections are used [26]. They copy features from the network's contracting part to its expansive part, where they are either concatenated with or added element-wise to the features of the expansion part.

The final pixel-wise segmentation is achieved using N_{classes} convolutional kernels of spatial size 1.

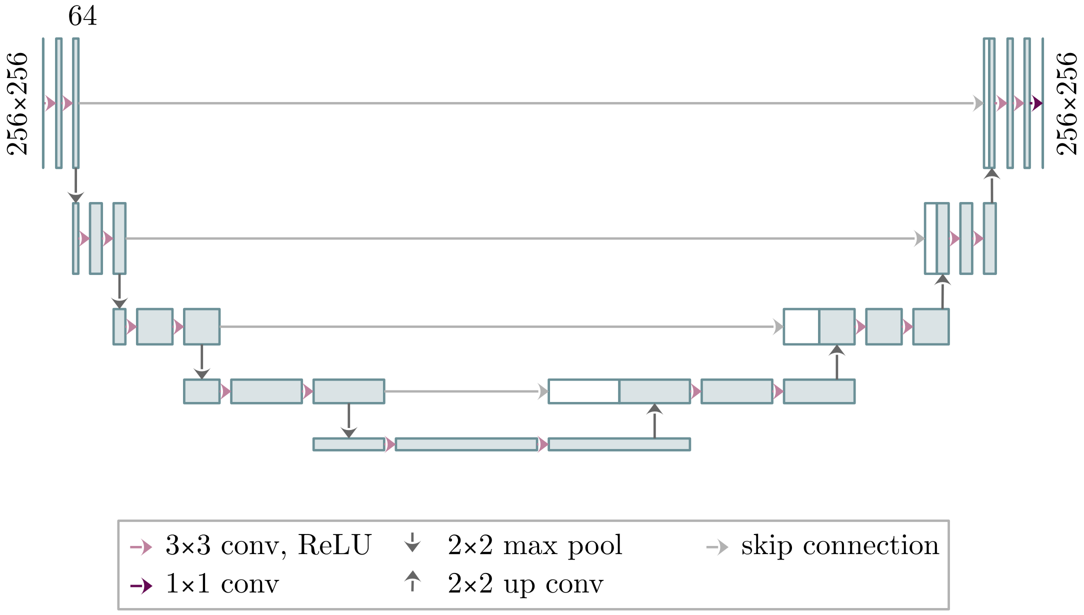


Figure 3.8: 2D U-Net architecture [107]. Within the contracting part (the encoder), the number of channels per layer (starting at 64) doubles in every stage, while the spatial size per dimension is halved (starting with an input image of 256×256 pixels). The opposite is happening in the expanding part (the decoder), where the spatial size increases by a factor of two from stage to stage while the number of channels decreases. Skip connections copy information from the encoder to the decoder.

3.2 Network Training and Inference

Many interesting tasks in computer vision and beyond can be formulated as optimisation problems

$$\min_{\mathcal{F}(x)} \mathcal{L}(\tilde{y} - \mathcal{F}(x)) \tag{3.16}$$

where we are interested in finding a function $\mathcal{F}(x)$ that should minimise the loss function $\mathcal{L}$, given inputs x and target outputs $\tilde{y}$.

As hinted at in Section 3.1, there are mathematical theorems that state that a very large class of functions – including the solutions to complicated optimisation problems – can be approximated using sufficiently deep or wide neural networks $\mathcal{F}_\theta$. These theorems and their proofs sadly do not provide any help in how the network parameters θ need to be chosen, in order to approximate the desired unknown function. Therefore, the big challenge in working with neural networks is to ensure both that a suitable network architecture is chosen and that the network's trainable parameters θ approach values where $\mathcal{F}_\theta$ is a good approximation of the ideal $\mathcal{F}$:

$$\theta = \operatorname{argmin} \mathcal{L}(\tilde{y} - \mathcal{F}_\theta(x)) \,, \tag{3.17}$$

Most solvers typically used for optimisation problems are computationally too expensive to compute for the vast parameter spaces of neural networks. In practice, comparatively basic first-order gradient descent algorithms are used to minimise the loss function. In the realm of deep learning, the term backpropagation is commonly used to refer to the efficient automated computation of first-order gradients using the chain rule in differential calculus [58, 74, 65].

Backpropagation is used to fit neural networks by computing the gradient of the loss function with respect to the network weights. Computing the gradient on the entire dataset at once is not practical or appropriate given the enormous datasets and high computational load. Stochastic gradient descent is employed instead [106, 12]. It computes the gradient of a small number of input-output pairs (one batch) at a time to obtain an estimate of the gradient. Bigger batch sizes allow for more accurate estimations of the actual gradient. In contrast, smaller batch sizes result in faster iterations since less computation and data IO operations are required. This, however, comes at the expense of a slower convergence rate [13]. The chosen batch size can be used to trade-off between the two effects. Because they are otherwise computationally costly, 3D networks tend to use tiny batch sizes, which can be as small as 1.

3.2.1 Loss functions

Loss functions between a network output $y = \mathcal{F}_\theta(x)$ and a target value $\tilde{y}$ are chosen depending on the task and are essential to steer the optimisation process. Below are examples of frequently used loss functions.

Mean squared error (MSE) loss

The mean squared error loss penalises deviations of the prediction y from the target $\tilde{y}$ by summing up the squared differences of every pixel p of the output P.

$$\mathcal{L}_{MSE}(y, \tilde{y}) = \frac{1}{P} \sum_{p \in P} (y - \tilde{y})^2 \tag{3.18}$$

The MSE loss is mainly used in regression tasks.

Cross entropy loss

The cross-entropy loss is usually applied just after applying a softmax function to the network output x.

For ease of notation, we provide the loss definition for the classification case, i.e. a segmentation case for an image with only a single pixel, i.e. $x \in \mathbb{R}^{1\times|C|}$ for classes $c \in C$.

The softmax operation takes into account all network outputs and turns them into members of a discrete probability distribution. For a classification network output of $|C|$ classes, the i-th class has a softmax output of

$$\text{softmax}(x_i) = \frac{\mathrm{e}^{x_i}}{\sum_{c\in C} \mathrm{e}^{x_c}} , \tag{3.19}$$

As expected from a probability distribution, all values will be between 0 and 1, and the sum of the softmax of all classes $c \in C$ adds up to 1.

The cross-entropy loss then is defined as

$$\mathcal{L}_{\text{CE}}(x) = -\log(\text{softmax}(x_{\tilde{c}})) , \tag{3.20}$$

with $\tilde{c}$ being the target class.

The cross-entropy loss then is defined as

$$\mathcal{L}_{\text{CE}}(x) = -\log(\text{softmax}(x_{\tilde{c}})) , \tag{3.21}$$

with $\tilde{c}$ being the target class.

In the case of segmentation, instead of classification, the softmax and consequent cross-entropy are computed for every pixel individually. The sum or mean of all accrued pixel-wise loss values can then be used to compute the total segmentation loss.

Dice Loss

The Dice loss [80] is commonly used in semantic segmentation on its own or in conjunction with the cross-entropy loss. It is based on the Dice similarity coefficient (DSC) metric, which will be discussed in Section 4.1.1.

$$\mathcal{L}_{DSC} = -\frac{2}{|C|}\sum_{c\in C}\frac{\sum_n \tilde{\mathbf{y}}_{c,n} s_{c,n}}{\sum_n(\tilde{\mathbf{y}}_{c,n} + s_{c,n})} \tag{3.22}$$

with $\tilde{\mathbf{y}}_{\mathbf{c,n}}$ the one-hot encoded target label where n denotes the location and c the class. $s_{c,n}$ the softmax output.

3.2.2 Optimizers

First-order gradient-based methods are widely used to solve the optimisation problem Equation (3.17). As explained above, in neural network training, the gradient is usually only an estimate obtained using stochastic backpropagation. Most popular optimisers build on top of the basic gradient descent or steepest descent algorithm, which updates the network parameters θ at every iteration step t with the gradient direction $(\nabla_\theta \mathcal{L})_t$:

$$\theta_{t+1} = \theta_t - \lambda \left(\nabla_\theta \mathcal{L}\right)_t \ , \tag{3.23}$$

where λ denotes the step size or learning rate. The choice of an appropriate learning rate is non-trivial and essential in order to escape local minima and finally converge to a solution. For that reason, several improvements to the basic gradient descent algorithm have been proposed. These enhanced algorithms perform well on various tasks, but there is no all-purpose universal optimiser [139].

Momentum

It has been known for some time that valleys in optimisation surfaces impede convergence because they force first-order gradient optimisers into an inefficient up-hill and down-hill jitter instead of traversing the valley. The momentum approach [99] is one way to address this problem and accelerate convergence by forcing the iteration step to take the direct way along the valley. Informally, the momentum M is the average of the last few gradient directions. The parameter update is then a combination of the gradient direction and the momentum.

$$\theta_{t+1} = \theta_t - \beta \left(\nabla_\theta \mathcal{L}\right)_t + \gamma M_t \tag{3.24}$$

The weighting constants β and γ steer the contribution of the momentum to the parameter update.

RMS-Prop

RMS-Prop [45] uses a running average r of the squared previous gradient directions, also called the second order momentum. The average decreases exponentially to guarantee that iterations from the distant past do not significantly influence parameter updates. It is otherwise conceptually similar to the basic momentum method.

$$r_{t+1} = \rho\, r_t + (1-\rho) \left(\nabla_\theta \mathcal{L}\right)_t \odot \left(\nabla_\theta \mathcal{L}\right)_t \tag{3.25}$$

$$\theta_{t+1} = \theta_t - \epsilon \frac{1}{\sqrt{r_{t+1}}} \odot \left(\nabla_\theta \mathcal{L}\right)_t \tag{3.26}$$

Here, ϵ is the global learning rate, $\odot$ is the element-wise multiplication, and ρ is a weighting constant.

Adam

Adam ”adaptive moments” [59] is one the most commonly used optimizers in semantic segmentation. It incorporates both the standard s and the second order momentum r (both decaying over time) and adds a bias correction for both terms, i.e., $\hat{s}$, and $\hat{r}$. The decay factors are denoted β_1, and β_2. The whole optimisation step is then computed as follows:

$$s_{t+1} = \beta_1 s_t + (1 - \beta_1)\left(\nabla_\theta \mathcal{L}\right)_t \tag{3.27}$$

$$r_{t+1} = \beta_2 r_t + (1 - \beta_2)\left(\nabla_\theta \mathcal{L}\right)_t \odot \left(\nabla_\theta \mathcal{L}\right)_t \tag{3.28}$$

$$\hat{s}_{t+1} = \frac{s_t}{1 - \beta_1^{t+1}} \tag{3.29}$$

$$\hat{r}_{t+1} = \frac{r_t}{1 - \beta_2^{t+1}} \tag{3.30}$$

$$\theta_{t+1} = \theta_t - \epsilon \frac{\hat{s}_{t+1}}{\sqrt{\hat{r}_{t+1}}} \tag{3.31}$$

3.2.3 Validation

Large quantities of trainable parameters make deep neural networks susceptible to overfitting. It is typical, therefore, not to use all data for network training but rather to keep some data separate for validation. Validation is performed throughout the training of a neural network to determine how well the learned model performs on inputs it has not been trained on. If no more improvements are observed on the validation data, the training process is said to have converged, and the training is usually stopped. The final results are computed on a third hold-out set called the test set to avoid overfitting effects stemming from the validation data.

If little data is available, this strategy might be changed to cross-validation. In cross-validation, the training process is repeated several times, each time with different portions of the dataset used for training, validation, and test. The test sets should be distinct among these repetitions and as big as possible to maximise the significance of the evaluation.

3.2.4 Inference

Once a network has been trained, i.e., its parameters θ have been fixed, it can be utilised repeatedly for its intended purpose. Unlike training, only a single forward pass across the network is necessary for inference. The gradient computation and subsequent parameter updates are no longer performed. As a result, inference requires a tiny amount of time compared to training. This makes neural networks ideal for scenarios where new data become available often and must be evaluated rapidly.

The inference of each voxel within a multi-class segmentation task is conducted by choosing the class $c \in C$, which has the highest softmax activation for the given voxel:

$$y = \underset{c \in C}{\operatorname{argmax}}(\operatorname{softmax}(x_c)), \tag{3.32}$$

3.2.5 Data augmentation

Small datasets often do not contain data in their full variety of possible poses and lighting conditions. Datasets are artificially extended to improve the robustness of a trained model, i.e., to encourage success for inputs slightly outside the input distribution. Data augmentation is the term commonly used to denote these transforms of the input data. Data augmentation is done under the premise that additional information can be derived from the original dataset by augmenting it.

Common data augmentation strategies include spatial transformations such as shifts, rotations, and scaling [129]. Furthermore, adding noise [83], or variations in colour, brightness, and contrast are employed to various degrees [83]. More advanced data augmentation strategies include mix-up [146], elastic deformations[21], GAN based augmentation [36], and neural style transfer[39, 96].

Chapter 4

Segmenting Distinct Bones

This chapter gives a more detailed introduction to the segmentation task itself in Section 4.1. We then focus on the challenges and potential solutions specific to distinct bone segmentation in Section **??**.

4.1 Segmentation

Segmentation is the process of assigning a label to every pixel or voxel of a digital image. Its most basic form – the binary segmentation – distinguishes only between two classes: Objects of interest in the foreground and the background. If precisely one out of $|C| > 2$ classes is assigned to every pixel, the term *multi-class* segmentation is used. This can easily be confused with *multi-label* segmentation, where multiple classes can be assigned to a single pixel, which often makes sense when classes are hierarchical. In our work, we will only use the first concept, multi-class segmentation.

A segmentation can be represented by its boundaries or by a label map, which assigns a label key – usually 0 for the background and positive whole numbers for the remaining labels – to each pixel depending on the label the pixel belongs to. In the case of a multi-class segmentation with $|C|$ possible classes, it can be helpful to transform the labelmap to its one-hot encoded form

$$\mathcal{O} : \mathbb{N}_{0\ldots|C|}^{x\times y\times z\times 1} \longrightarrow \mathbb{N}_{0,1}^{x\times y\times z\times |C|}, \tag{4.1}$$

which has an additional dimension along which $|C|$ binary segmentations are stacked.

Segmentation of medical images assigns information regarding anatomical structure or pathological status to individual pixels. Among others, it thus enables the identification of regions of interest, for example for quantifying tumour sizes and their change over time, for studying anatomical structures, and for supporting treatment planning and navigation in radiation therapy and surgery [119].

Segmentations can be created manually by medical professionals, automatically by computers, or by a combination of the two. Automatic segmentation is a difficult task in medical imaging because medical images are complex and lack simple linear features. Further difficulties arise due to partial volume effects, imaging artefacts, noise, and the similarity of greyscale values among different soft tissues [119], see also Section 2.3.

4.1.1 Segmentation metrics

The quality of a segmentation can be validated using other segmentations as a comparison. In medical image segmentation, the manual segmentation generated by medical professionals is usually considered the target against which the quality of a predicted segmentation is assessed.

Spatial overlap based metrics

A segmentation can be viewed as a pixel-wise classification. In the case of multi-class segmentation with N labels, the evaluation can be split into N binary classifications using a One-vs.-rest strategy.

After binary classification of a single pixel, four different outcomes are possible that are presented in Table 4.1. In segmentation tasks, every pixel is assigned one of the four outcomes by comparing the predicted segmentation to the target segmentation, as illustrated in Figure 4.1. The total counts of pixels per outcome can then be used to compute overlap-based segmentation metrics [128].

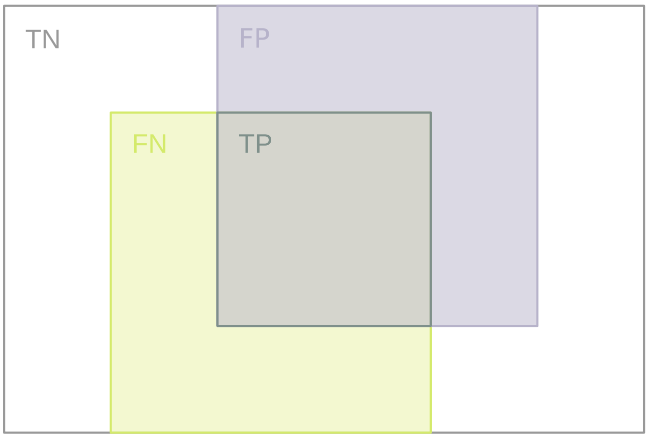


Figure 4.1: Areas of TP, TN, FP, and FN voxels when a prediction (purple) and a ground truth (green) partially overlap.

Table 4.1: Confusion matrix.

	Actual positive	Actual negative
Predicted positive	true positive (TP)	false positive (FP)
Predicted negative	false negative (FN)	true negative (TN)

Some of the most commonly encountered overlap metrics are:

- **Recall = sensitivity = true positive rate (TPR):**

  $$TPR = \frac{TP}{TP+FN}$$

  Recall is the fraction of relevant instances that were found.

- **Precision = positive predictive value (PPV):**

  $$PPV = \frac{TP}{TP+FP}$$

  Precision is defined as the proportion of relevant instances found among the retrieved instances.

- **Dice similarity coefficient (DSC)**:

  $$DSC = \frac{2TP}{2TP+FP+FN}$$

  The DSC is mathematically equivalent to the F_1 score which consists of the harmonic mean of recall and precision. The DSC metric is often used in practice to assess the quality of segmentation algorithms. It can further be used to create a corresponding DSC loss function, see Section 3.2.1.

- **Jaccard index (JAC)**:

  $$JAC = \frac{TP}{TP+FP+FN}$$

  The JAC is related to the DSC by

  $$DSC = \frac{2JAC}{1+JAC}$$

  and therefore does not provide additional information over the DSC [128].

To generalise this measure in a multi-class setting, it is possible to use the mean over all classes $c \in C$ [33]

$$DSC_{mean} = \frac{1}{|C|}\sum_{c\in C} DSC_c$$

or to use a generalised version of the multi-class Dice score [24, 33]:

$$DSC_{multi} = \frac{2\sum_{c\in C}\alpha_c TP_c}{\sum_{c\in C}\alpha_c l(2TP_c + FP_c + FN_c)}, \tag{4.2}$$

where the parameter $\alpha_c S$ can be used to weight the classes.

Spatial distance based metrics

In contrast to overlap metrics, distance-based metrics operate not on voxel volumes but point sets or surfaces. They are useful to ensure an accurate boundary delineation [128, 47].

Hausdorff distance (HD):

$$HD = \max\left(h(A,B), h(B,A)\right), \tag{4.3}$$

for two finite point sets A and B, where

$$h(A,B) = \max_{a\in A} \min_{b\in B} ||a-b|| \tag{4.4}$$

is the definition of the directed Hausdorff distance. A common choice for the norm $||\cdot||$, is the Euclidean L_2 norm. Because the HD is very sensitive to outliers, it is common to use the q^{th} quantile – often 95^{th} – instead of the maximum of the distances. The HD is based on pairwise distances between all points in two point clouds. It is therefore computationally very intensive to find the HD of segmentation boundaries given by many voxels [128].

4.1.2 Segmentation methods

There exist a plethora of possible segmentation algorithms in general and for bone segmentation in particular. They range from manual to fully automated once set up and everything in between. In the following, we provide a rough overview only. For a comprehensive list of algorithms used for bone segmentation in particular, we direct the interested reader to Sjoquist [122].

Manual

Manual segmentations are performed by delineating or filling in all regions of interest in an image. In the case of three-dimensional images, the segmentation has to be performed slice by slice, making this a very lengthy process [29]. Performing manual segmentations requires prior medical knowledge about the structures of interest and their representation in the chosen image modality. Depending on the structure of interest, drawing a clear boundary might be difficult, and only an autopsy might reveal the segmentation's ground truth. For these reasons, the results of manual segmentations differ among experts (inter-rater variability) and even among repeated attempts of the same expert (intra-rater variability).

Semi-Automatic

Interactive segmentation tools such as 3D Slicer [31], ITK-Snap [143], and others [144, 71] reduce the time needed for segmentation by providing a wide array of algorithms that reduce the time needed for the segmentation process. These semi-automatic algorithms require user interventions such as setting seeds or choosing an area of interest. The algorithms range from comparatively simple, such as thresholding[123], and edge-detection [94] to intricate, such as graph cuts [14].

Bone tissue has a very distinct HU value (see also Section 2.3), which facilitates distinguishing bone tissue from other tissue types and hence aids segmentation. A straightforward approach to segmentation is, therefore, thresholding.

Thresholding can be performed globally on a whole image or locally. It divides an image into different parts using the valleys and peaks in the intensity histogram of an image [89]. The thresholds are chosen either visually or according to the expected HU values of the tissue of interest. Thresholding is very fast but requires the structures of interest to have distinct intensity levels and is not robust to noise [89].

Noise, artefacts, and the lower HU values of cancellous bone hinder the sole use of global thresholds to obtain high-quality segmentation results for bone tissue [97]. If only a low number of segmentations needs to be performed, interactive medical tools [71, 143, 144, 31] can be used to set multiple thresholds, or adaptive thresholding algorithms can be used [98, 17, 43, 147, 104]. This can remedy the situation but still requires multiple, manual steps, which make these approaches more cumbersome and hard to scale up [7].

Edge-based segmentation works by detecting discontinuities in the intensity values between image regions using discretised first- and second-order derivative kernels. The thus detected edges are combined into an edge chain. Thresholding is used to remove any false or weak edges. The edge detection and rejection is repeated with different threshold values until closed boundaries are found [102, 62, 119, 42].

Region growing is an iterative process that starts with one or several user specified seeds. In each iteration, the algorithm probes the region's neighbourhood and adds those pixels that satisfy a predefined similarity criterion [147, 34, 54].

K-nearest-neighbours (knn) is an unsupervised clustering algorithm. The goal is to partition the data into K clusters. Data points are iteratively assigned to the nearest cluster, and then the clusters themselves are updated. This procedure is sensitive to the number of clusters K, the primary initialisation, and local minima [119].

Graph cuts are a fast efficient representation for solving complex energy functionals that segment an image. As a first step, the image needs to be converted to a graph. Each pixel serves as a node, and vertices connect adjacent nodes. Each node's weight represents the similarity of the nodes it connects. After adding a sink and a target node, max-flow min-cut graph algorithms can be used to divide the graph into two sections and thus segmenting the image [14].

Active contour or snake algorithms start with a user-defined contour refined iteratively by minimising the contour's energy function. This function contains internal energy terms to control the smoothness of the contour and external energy terms that attract the contour towards the object of interest [22, 57].

In practice, many of the above approaches are combined and can lean more towards the manual or the automated side. Most algorithms target specific body parts and selected bones and are unsuitable for full-body distinct bone segmentation.

Automated

Fully automatic segmentation methods do not require user inputs and thus save time and potentially costs.

Atlas-based segmentation only works for anatomical structures that always appear in roughly the same location with a reasonably similar structure; it can not be applied to tumour detections and comparable tasks. The technique is primarily used in brain region segmentation [1]. It works by registering a template brain with its corresponding known labelmap, the so-called atlas, to the input image. The segmentation can then be read out from the warped template. To improve the results, average atlases or multiple atlases with a decision fusion strategy can be used [50]. A comparison of different atlas-based approaches for binary bone tissue segmentation can be found in [5].

Statistical shape models can be used to segment single bones or organs. All training data is registered in the first step to achieve point-wise correspondence. Then, the main modes of shape variations can be determined using statistical methods and dimension reduction, such as a principal component analysis (PCA). Once the shape model has been built, it can be fitted to an image to segment the structure [44, 115, 109, 101].

One approach that has been specifically suggested for full-body bone-tissue segmentation uses a bottom-up approach to generate supervoxels with a watershed algorithm. After recursive supervoxels growing and merging, the authors formulate their problem as a binary conditional random field optimisation problem over the graph of supervoxels and solve it using a support vector machine (SVM) [78].

Most of the recent works on full-body bone-tissue segmentation have focused on convolutional neural networks to work on automated bone-tissue segmentation and have outperformed the threshold-based approaches [81, 60, 88]. The prevalent approach is to use 2D axial slices of CT scans to conduct the segmentation using 2D U-Nets [107, 49] and a fully supervised approach. Noguchi et al. [88] use an in-house dataset of 32 scans to train their models and achieve DSC scores of up to 0.98, basically solving the case of bone-tissue segmentation.

4.1.3 Prior work in distinct bone segmentation

In contrast to bone-tissue segmentation, distinct bone segmentation distinguishes bones not only from other tissues but individual bones from one another. Numerous works segment one particular bone only, using a wide variety of the solutions sketched above. Some examples include the skull [56, 81], mandible [127, 130], femur [56, 136], tibia and ulna [38], and the scapula [127].

Vertebra segmentation is a well-studied task that involves classifying many individual bones. Most recent approaches lean towards neural network-based solutions [9, 116, 51, 68]. Another task involving many bones of the same group is rib segmentation [140, 15, 76, 142]. In both rib segmentation and vertebra segmentation, individual instances of one group of bones are detected. As such, many approaches include post-processing steps or location priors that are not easily transferable to a broader distinct bone segmentation task.

To our knowledge, only an handful of published papers on human full-body or half-body bone segmentation exists [78, 60, 88, 70, 11, 37, 73]. Of those publications only three [11, 37, 73] attempt to perform *distinct* bone segmentation.

Bieth et al. [11] use Haar-like features and geometric features within an iterative random forest approach. They segment iteratively finer structures and use the previously obtained centroids as landmarks to guide the following iterations. There is no discussion on which exact bones they are segmenting, but their maximum number of labels is reported as 88. An evaluation on their dataset of 20 whole-body CT scans of healthy subjects with a $2.6\,\mathrm{mm} \times 2.6\,\mathrm{mm} \times 2\,\mathrm{mm}$ resolution results in a DSC of 84.2 ± 6.5.

Fu et al. [37] use multiple thresholds to locate bone voxels and then fit a hierarchical atlas to distinguish 62 individual bones. They still perform some manual steps to guide the registration. Their method runs on approximately $1\,\mathrm{mm}$ resolution CT data of 19 patients with a mean DSC of 0.91. Sjoquist [122] re-implement the method without requiring manual steps and use the results as a step in their pipeline to locate metastatic bone disease.

Lindgren Belal et al. [73] use a two-stage approach to segment 49 distinct bones from $3.27\,\mathrm{mm} \times 3.27\,\mathrm{mm} \times 3.75\,\mathrm{mm}$ resolution PET/CT scans. In a first step, they use a CNN which outputs landmark location. The landmarks include rib joints and vertebral processes, which are not yet assigned to any specific bone. Identification of the matching bones is then made using an active shape model. Another network detects rib centre lines. In a second step, the vertebra landmarks and the rib centre lines are fed into a segmentation network together with the original image. Unlike many other neural network-based approaches, they do not use a U-Net structure but a network structure that uses convolutions on three different scales of the input, all merged simultaneously. The receptive field is increased using dilated convolutions. They use a training set of 100 subjects and validate on 46 subjects.

4.2 Challenges

There are several challenges particular to the automated segmentation of distinct bones. The most prevalent of these challenges and a few potential solutions are given below.

4.2.1 Class abundance and imbalance

In most applications of medical image segmentation, only a handful of classes are segmented at once. We work on up to 125 bone classes in our distinct bone segmentation task at once. This is a computational issue mostly when evaluating the loss function. When using the DSC loss (3.2.1), the necessary one-hot computation (see Equation (4.1)) of a 3D volume with 125 classes gets very big very quickly. Even when using loss functions that do not require the one-hot encoding, e.g., cross-entropy loss (see Section 3.2.1), the gradients in relation to all classes need to be computed, which can be a computationally heavy operation.

Bones also come in very different shapes and sizes, as illustrated in Section 2.1. The difference in size also manifests in the number of voxels that any given label has in a final segmentation. Big bones such as the femurs, coxae, or the skull consist of well over 10’000 foreground voxels, while the small bones of the wrist consist of fewer than 100 voxels.

In machine learning, class imbalance can have detrimental effects on the trained models. In the simplest case of binary classification, a trained classifier might consistently predict the most prevalent class, irrespective of the input, having learned that guessing the prevalent class leads to good loss values. In segmentation, the same effect might lead to pixels primarily being assigned the most prevalent class. In medical image segmentation, this is almost always the background label.

Different strategies are possible to remedy class imbalance, whether data-based or algorithm-based. We provide an example for each category.

Balanced data sampling

In semantic segmentation of medical images, more often than not, the objects of interest only make up a tiny fraction of an image. In contrast, the background areas are often huge, look redundant and tend to be easily detectable [75]. One possibility to work around this issue is to sample the network inputs so that rare classes appear more often than they naturally would. This strategy works very well in conjunction with 3D input data, where the network images need to be cropped in any case because of memory restraints. It also works well in conjunction with a random cropping data augmentation scheme. Different schemes are possible such as [49] who sample two thirds of patches at random, while they require the remaining third to contain at least one foreground voxel. Another approach is selective sampling, where inputs that the network previously made errors are sampled more often [133]. We examine the effect of balanced sampling in our publication in Chapter 5.

Balancing Loss Functions

Several loss functions have been proposed specifically to tackle class imbalance [126].

Weighted Cross-Entropy was famously used in [107] and is given by the following formula:

$$\mathcal{L}_{\mathrm{WCE}}(y,\tilde{y}) = -w(\tilde{y}) \log\left(s_{\tilde{y}}\right) \tag{4.5}$$

with $\tilde{y}$ being the target class, $s_{\tilde{y}}$ the softmax output of the target class and $w(c)$ being a weight function that assigns a weight to each class c, usually more weight to minority classes, such as the inverse of the number of pixels of that class P_c i.e. $w(c) = \frac{N-P_c}{P_c}$, where N is the total number of pixels of the image. The weight function is usually chosen upfront and treated as a hyperparameter instead of being learnt.

The **Generalised Dice Loss** builds on the generalised dice metric, which has been proposed by [24]. It has been first discussed as a loss function by [126].

$$\mathcal{L}_{\mathrm{GDSC}} = 1 - 2\frac{\sum_{c=0}^{C-1} w(c) \sum_n \tilde{y}_{c,n} s_{c,n}}{\sum_{c=0}^{C-1} w(c) \sum_n (\tilde{y}_{c,n} + s_{c,n})}\,, \tag{4.6}$$

where n iterates over all spatial elements of the network output. The weights are chosen to ensure the contribution of each label inverse to its volume $w(c) = 1/\left(\sum_n \tilde{y}_{c,n}\right)^2$. In-depth comparisons in [126] have shown a slight benefit of the generalised over the standard dice loss metric using the U-Net architecture and a general advantage of overlap-based loss functions - such as $\mathcal{L}_{\mathrm{DSC}}$ and $\mathcal{L}_{\mathrm{GDSC}}$ - over cross-entropy loss functions.

4.2.2 Data scarcity

Supervised learning requires many fully labelled samples to train machine learning networks. In medical applications, it is often challenging to collect enough data. To solve this issue, different paths are possible.

Active learning

Active learning tries to maximise the usefulness of annotator time by choosing the subsequent to-be-annotated data such that it leads to the best model improvement [117, 25, 16]. In order to assess the model improvement gained by adding a new sample, the sample's informativeness has to be estimated. This is usually done by computing the uncertainty of a prediction, arguing that annotating and adding samples with a high uncertainty leads to a high information gain of the trained model. We evaluated this approach in Chapter 6. An alternative option is representativeness, where the sampling of new data from different areas of the data distribution is encouraged.

Data augmentation

Data augmentation is a standard way to increase the robustness of the model. Augmentations can take many forms, but they fall primarily into two categories: geometric transformations and variation in visual representation. More details can be found in Section 3.2.5. We evaluate the use of data augmentation in Chapter 5.

Weakly-supervised learning

Weakly-supervised learning allows learning segmentation without access to pixel-wise segmentation labels. It is used for anomaly detection [138], brain tumours [52] and multiple sclerosis lesions [2]. A classification label usually replaces the pixel-wise segmentation labels, and generative models are used to create data in one class or another. Difference maps can then be used to create segmentations. Changing this approach from pathology to anatomical segmentation in a multi-class setting is not trivial.

All in all, automated distinct bone segmentation is a challenging task. The following publications will discuss our solutions to these challenges in more detail.

Chapter 5

3D Segmentation Networks for Excessive Numbers of Classes: Distinct Bone Segmentation in Upper Bodies

The publication presented in this chapter covers our first successful attempt at upper-body distinct bone segmentation. It elaborates where the challenges of this task lie and presents a 3D U-Net type architecture that managed to achieve first baseline results. We also compared a 2D to a 3D approach, concluding that only 3D was suitable for our case. We ablated several components of our approach, specifically the input size, the data sampling scheme, and the loss function. We compared our approach to other published results with encouraging outcome: despite distinguishing more classes and having a much smaller dataset we achieved competitive accuracies and faster inference times.

Publication. The following manuscript was presented at the *International Workshop on Machine Learning in Medical Imaging* (MLMI) in conjunction with the *23rd International Conference on Medical Image Computing and Computer-Assisted Intervention* (MICCAI), October 2020, Lima, Peru, held virtually. It was published as part of the workshop proceedings[1] [111] .

© Springer Nature. Reprinted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, 3D Segmentation Networks for Excessive Numbers of Classes: Distinct Bone Segmentation in Upper Bodies, Eva Schnider, Antal Horváth, Georg Rauter et al. (2020).

[1] https://doi.org/10.1007/978-3-030-59861-7_5

3D Segmentation Networks for Excessive Numbers of Classes: Distinct Bone Segmentation in Upper Bodies

Eva Schnider[1](✉), Antal Horváth[1], Georg Rauter[1], Azhar Zam[1], Magdalena Müller-Gerbl[2], and Philippe C. Cattin[1]

[1] Department of Biomedical Engineering, University of Basel, Allschwil, Switzerland
{eva.schnider,antal.horvath,georg.rauter,azhar.zam,m.mueller-gerbl, philippe.cattin}@unibas.ch

[2] Department of Biomedicine, Musculoskeletal Research, University of Basel, Basel, Switzerland

Abstract. Segmentation of distinct bones plays a crucial role in diagnosis, planning, navigation, and the assessment of bone metastasis. It supplies semantic knowledge to visualisation tools for the planning of surgical interventions and the education of health professionals. Fully supervised segmentation of 3D data using Deep Learning methods has been extensively studied for many tasks but is usually restricted to distinguishing only a handful of classes. With 125 distinct bones, our case includes many more labels than typical 3D segmentation tasks. For this reason, the direct adaptation of most established methods is not possible. This paper discusses the intricacies of training a 3D segmentation network in a many-label setting and shows necessary modifications in network architecture, loss function, and data augmentation. As a result, we demonstrate the robustness of our method by automatically segmenting over one hundred distinct bones simultaneously in an end-to-end learnt fashion from a CT-scan.

Keywords: 3D segmentation · Deep learning · Many label segmentation

1 Introduction

The segmentation of distinct bones from CT images is often performed as an intermediate or preprocessing task for planning and navigation purposes to provide semantic feedback to those systems. It is also crucial for the evaluation of the progress of bone diseases [7], or for the quantification of skeletal metastases [17]. In Virtual Reality (VR) tools [5,14], the distinct segmentation of bones permits more fine-grained control over rendered body parts and can serve an educational purpose by teaching skeletal anatomy. Due to its distinctive high Hounsfield unit (HU) values in CT images, cortical bone tissue can be segmented approximately using thresholding. However, random intensity variations and the relatively low HU value of squamous bones hinder accurate results [18].

© Springer Nature Switzerland AG 2020

M. Liu et al. (Eds.): MLMI 2020, LNCS 12436, pp. 40–49, 2020.
https://doi.org/10.1007/978-3-030-59861-7_5

For a precise segmentation, or the separation of individual bones, more elaborate methods are needed. For the analysis and segmentation of single bones, statistical shape or appearance models are applied [19,21,22]. For whole skeletons, atlas segmentations using articulated joints have been used in mice [1], and for human upper bodies [7]. A combination of shape models and convolutional neural networks (CNN) have been employed in [17] to segment almost fifty distinct bones. Their multi-step approach consists of an initial shape model corrected landmark detection, followed by a subsequent voxel-wise segmentation. Solely CNN based methods have been used for full-body bone tissue segmentation, without labelling of individual bones [13], and for segmentation of bones of groups, such as vertebrae [23]. To our knowledge, no simultaneous segmentation of all distinct bones of a human upper body by the use of CNNs has been published so far.

Fully automated methods driven by CNNs have shown great results for various tasks in medical image analysis. They excel at pathology detection [2,10,11] as well as at segmenting anatomical structures [9,16,20] for a wide array of body regions and in both 2D and 3D. In conjunction with data augmentation, good results have been reported even when training networks on as little as 1–3 fully annotated scans [3,4]. However, in typical 3D medical image segmentation tasks, distinctions are made for a handful or up to a dozen classes. Many established methods developed for a few classes fail when dealing with the over hundred classes for our particular case, or are not practical anymore due to restrictions in computational time and memory.

In this work, we present, which kinds of preprocessing, network choice, loss function and data augmentation schemes are suitable for 3D medical image segmentation with many labels at once, using the example of distinct bone segmentation in upper bodies. Our contributions are: 1) We discuss essential adaptions concerning network choice and data augmentation when performing 3D segmentation in a many-label setting. 2) We examine different sampling strategies and loss functions to mitigate the class imbalance. 3) We present results on a 3D segmentation task with over 100 classes, as depicted in Fig. 1.

2 Methods

Segmenting many classes simultaneously in 3D comes at a cost in computational space and time. In the following, we discuss how this affects and limits not only

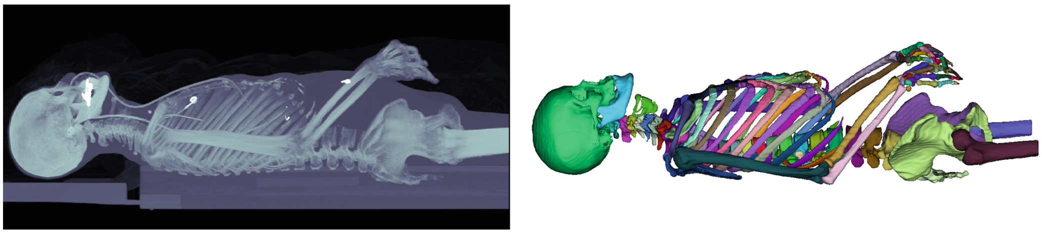
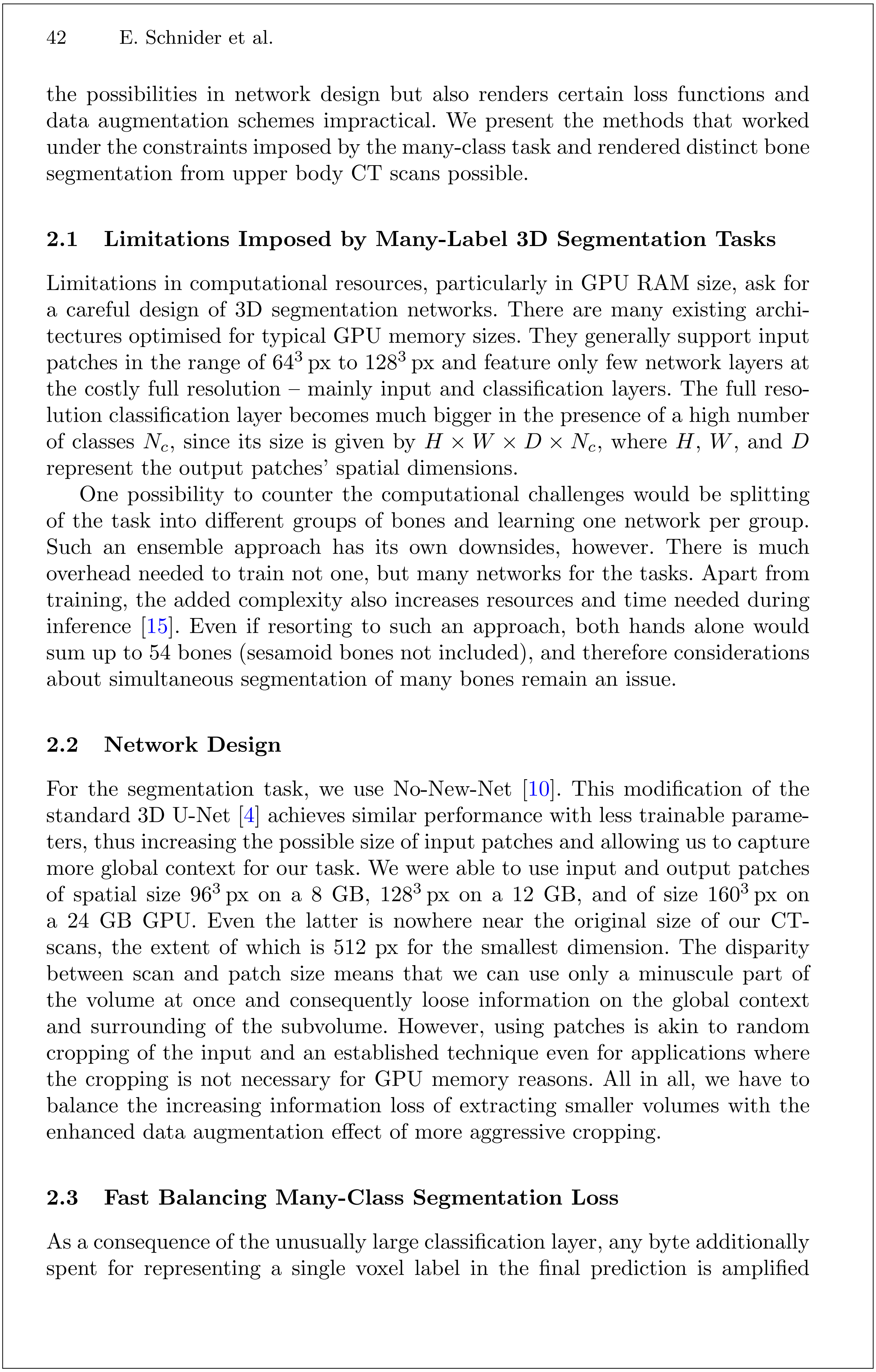


Fig. 1. Left: maximum intensity projection of one of our upper body CT-scans. Right: the manual target segmentation depicting 125 different bones with individual colours. (Color figure online)



the possibilities in network design but also renders certain loss functions and data augmentation schemes impractical. We present the methods that worked under the constraints imposed by the many-class task and rendered distinct bone segmentation from upper body CT scans possible.

2.1 Limitations Imposed by Many-Label 3D Segmentation Tasks

Limitations in computational resources, particularly in GPU RAM size, ask for a careful design of 3D segmentation networks. There are many existing architectures optimised for typical GPU memory sizes. They generally support input patches in the range of 64^3 px to 128^3 px and feature only few network layers at the costly full resolution – mainly input and classification layers. The full resolution classification layer becomes much bigger in the presence of a high number of classes N_c, since its size is given by $H \times W \times D \times N_c$, where H, W, and D represent the output patches' spatial dimensions.

One possibility to counter the computational challenges would be splitting of the task into different groups of bones and learning one network per group. Such an ensemble approach has its own downsides, however. There is much overhead needed to train not one, but many networks for the tasks. Apart from training, the added complexity also increases resources and time needed during inference [15]. Even if resorting to such an approach, both hands alone would sum up to 54 bones (sesamoid bones not included), and therefore considerations about simultaneous segmentation of many bones remain an issue.

2.2 Network Design

For the segmentation task, we use No-New-Net [10]. This modification of the standard 3D U-Net [4] achieves similar performance with less trainable parameters, thus increasing the possible size of input patches and allowing us to capture more global context for our task. We were able to use input and output patches of spatial size 96^3 px on a 8 GB, 128^3 px on a 12 GB, and of size 160^3 px on a 24 GB GPU. Even the latter is nowhere near the original size of our CT-scans, the extent of which is 512 px for the smallest dimension. The disparity between scan and patch size means that we can use only a minuscule part of the volume at once and consequently loose information on the global context and surrounding of the subvolume. However, using patches is akin to random cropping of the input and an established technique even for applications where the cropping is not necessary for GPU memory reasons. All in all, we have to balance the increasing information loss of extracting smaller volumes with the enhanced data augmentation effect of more aggressive cropping.

2.3 Fast Balancing Many-Class Segmentation Loss

As a consequence of the unusually large classification layer, any byte additionally spent for representing a single voxel label in the final prediction is amplified millionfold. Using a dense representation of the prediction instead of a sparse one will tip the balance easily towards an out-of-memory error. We thus use sparse representations of the class-wise predictions and ground truths for computation of the loss. To counter the high imbalance in the number of voxels per class, we use the multi-class cross-entropy loss in conjunction with a Dice similarity coefficient (DSC) loss over all classes $c \in C$: We chose to use an unweighted linear combination of the two, following the implementation given in [10]:

$$\mathcal{L}_{\text{X-Ent + DSC}} \coloneqq \mathcal{L}_{\text{X-Ent}} + \sum_{c \in C} \mathcal{L}^{c}_{\text{DSC}}. \tag{1}$$

2.4 Resourceful Data Augmentation

We utilise various data augmentation techniques to increase the variety of data the network encounters during training. We use random sampling of the input patch locations in two flavours: Uniform random sampling returns every possible patch with the same probability. With balanced sampling, every class has the same probability of being present in the chosen subvolume. Balanced sampling results in high variability in the field of views of the (input) patches while asserting to repeatedly present all bones, even small ones, to the network.

Much like random cropping, many of the other prevalent techniques in 3D segmentation such as affine transformations, elastic deformations, and changes in brightness and contrast can be employed unhindered in the many-label setting. Contrarily, some augmentation schemes – notably MixUp [24] and its variants – work with dense labels and losses, thus causing tremendous inflation of the classification layer size and loss calculation time. We, therefore, omit the latter kind of data augmentation and concentrate on the first kind.

2.5 Implementation Details

Our experiments are built on top of the NiftyNet [8] implementation of the No-New-Net [10]. We modified the network architecture only in the number of channels of the classification layer, to account for the different amount of classes. We used the Leaky ReLU activation function with a leak factor of 0.02, and instance normalisation. In contrast to the No-New-Net publication [10], we were only able to fit a batch size of 1 due to the high memory demands of our many-class case. We optimised our networks using Adam [12] with a learning rate of 0.001 and ran 20 000 iterations of training.

3 Experiments

For lack of publicly available data sets with many-label distinct bone segmentation, our experiments are conducted on an in-house data set, consisting of five CT scans and their voxel-wise segmentation into 126 classes. To counter the low number of labelled images, we use 5-fold cross-validation throughout.

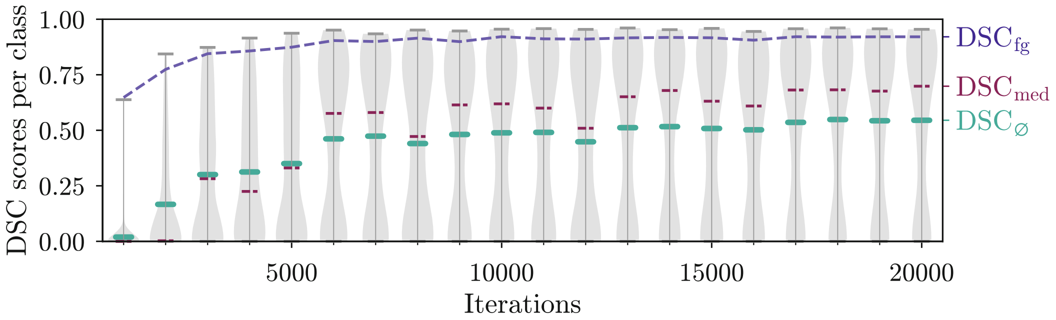


Fig. 2. Development of DSC scores (see subsect. 3.2) over the course of training. The distribution of per-class DSC scores is indicated by the violins (grey area). Additionally, the mean, median, and foreground DSC scores (3), are provided.

3.1 Data Set and Preprocessing

The five CT scans were taken and annotated by our university's anatomical department. The resulting voxel-wise segmentation consists of 126 classes – one for each kind of bone in the scans, plus background. The scans were taken from individual subjects aged 44–60, three of whom were female, two male. The field of view starts at the top of the skull and includes the area below until approximately mid-femur. All subjects lie on their backs, arms either resting on the lap or crossed over the stomach, a posture variation that makes the segmentation task harder. The size of each scan was $512 \times 512 \times H$, where the value of H ranges from 656 to 1001. In-plane resolutions vary from 0.83 mm × 0.83 mm to 0.97 mm × 0.97 mm while inter-plane spacing ranges from 1.0 mm to 1.5 mm.

To be able to capture more body context within an input patch, we resampled our data to 2 mm per dimension – approximately half the original resolution – resulting in volumes of $214 - 252 \times 215 - 252 \times 477 - 514$. We used bilinear interpolation for the scans and nearest neighbour interpolation for the label volume.

3.2 Evaluation

To evaluate the network's ability to correctly label and delineate each bone, we use the DSC of individual classes c in all our experiments: $\mathrm{DSC}_c = \frac{2|P_c \odot G_c|}{|P_c|+|G_c|}$, where P_c and G_c represent the pixel-wise binary form of the prediction of class c and the corresponding ground truth. To obtain a combined score for a whole group of bones over all cross-validation sets, we provide the median DSC. We furthermore provide the distance from the median to the upper and lower uncertainty bound, which correspond to the 16 and 84 percentile. If certain bones are not detected at all, i.e. their DSC equals 0, they are excluded to not distort the distribution. Instead, we provide the detection ratio

$$\mathrm{dr} := \frac{\#\text{ bones with DSC} > 0}{\#\text{ all bones}}. \tag{2}$$

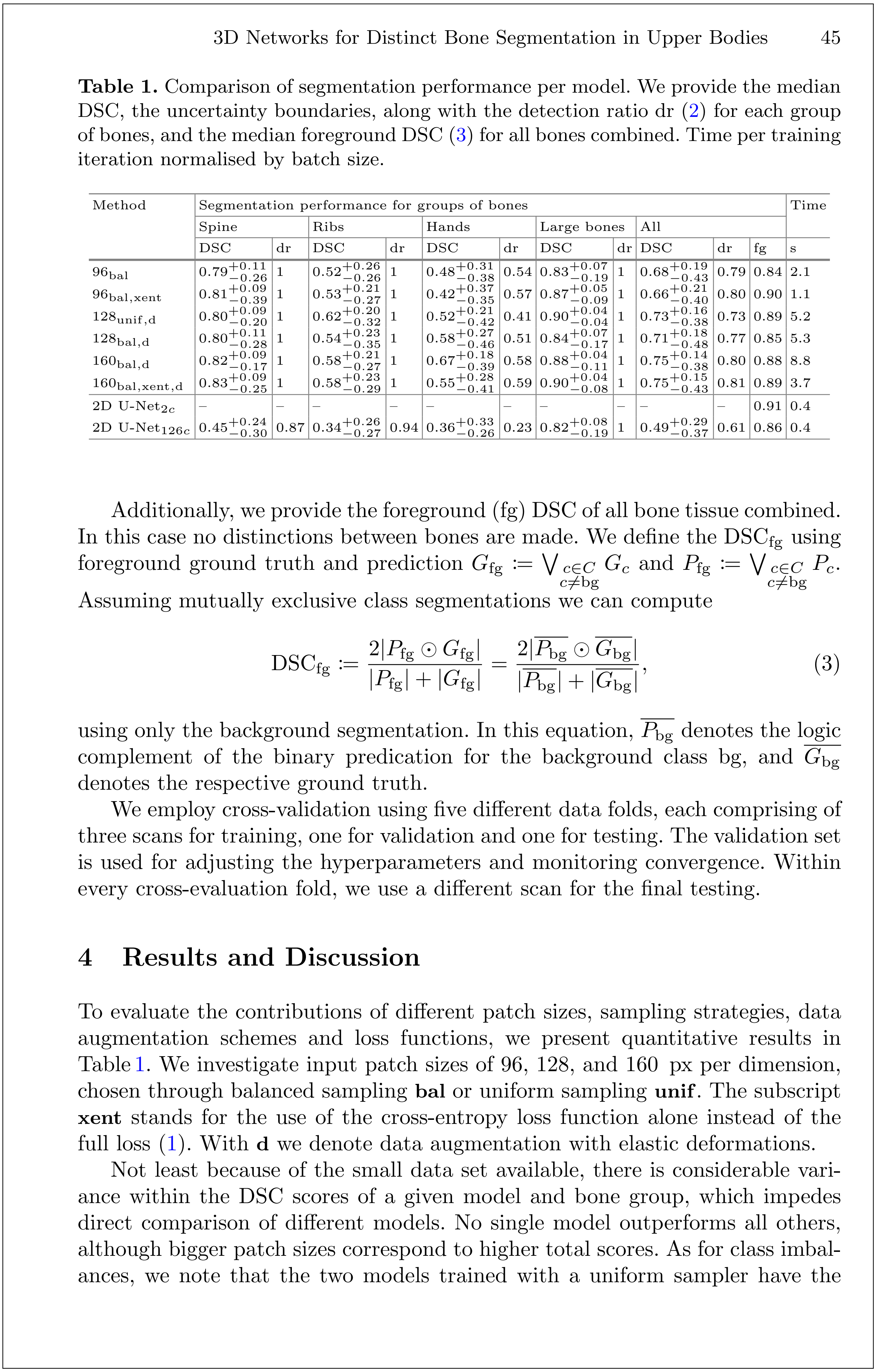


Table 1. Comparison of segmentation performance per model. We provide the median DSC, the uncertainty boundaries, along with the detection ratio dr (2) for each group of bones, and the median foreground DSC (3) for all bones combined. Time per training iteration normalised by batch size.

Method	Segmentation performance for groups of bones											Time
	Spine		Ribs		Hands		Large bones		All			
	DSC	dr	DSC	dr	DSC	dr	DSC	dr	DSC	dr	fg	s
96_{bal}	$0.79^{+0.11}_{-0.26}$	1	$0.52^{+0.26}_{-0.26}$	1	$0.48^{+0.31}_{-0.38}$	0.54	$0.83^{+0.07}_{-0.19}$	1	$0.68^{+0.19}_{-0.43}$	0.79	0.84	2.1
$96_{\text{bal,xent}}$	$0.81^{+0.09}_{-0.39}$	1	$0.53^{+0.21}_{-0.27}$	1	$0.42^{+0.37}_{-0.35}$	0.57	$0.87^{+0.05}_{-0.09}$	1	$0.66^{+0.21}_{-0.40}$	0.80	0.90	1.1
$128_{\text{unif,d}}$	$0.80^{+0.09}_{-0.20}$	1	$0.62^{+0.20}_{-0.32}$	1	$0.52^{+0.21}_{-0.42}$	0.41	$0.90^{+0.04}_{-0.04}$	1	$0.73^{+0.16}_{-0.38}$	0.73	0.89	5.2
$128_{\text{bal,d}}$	$0.80^{+0.11}_{-0.28}$	1	$0.54^{+0.23}_{-0.35}$	1	$0.58^{+0.27}_{-0.46}$	0.51	$0.84^{+0.07}_{-0.17}$	1	$0.71^{+0.18}_{-0.48}$	0.77	0.85	5.3
$160_{\text{bal,d}}$	$0.82^{+0.09}_{-0.17}$	1	$0.58^{+0.21}_{-0.27}$	1	$0.67^{+0.18}_{-0.39}$	0.58	$0.88^{+0.04}_{-0.11}$	1	$0.75^{+0.14}_{-0.38}$	0.80	0.88	8.8
$160_{\text{bal,xent,d}}$	$0.83^{+0.09}_{-0.25}$	1	$0.58^{+0.23}_{-0.29}$	1	$0.55^{+0.28}_{-0.41}$	0.59	$0.90^{+0.04}_{-0.08}$	1	$0.75^{+0.15}_{-0.43}$	0.81	0.89	3.7
2D U-Net$_{2c}$	–	–	–	–	–	–	–	–	–	–	0.91	0.4
2D U-Net$_{126c}$	$0.45^{+0.24}_{-0.30}$	0.87	$0.34^{+0.26}_{-0.27}$	0.94	$0.36^{+0.33}_{-0.26}$	0.23	$0.82^{+0.08}_{-0.19}$	1	$0.49^{+0.29}_{-0.37}$	0.61	0.86	0.4

Additionally, we provide the foreground (fg) DSC of all bone tissue combined. In this case no distinctions between bones are made. We define the DSC_{fg} using foreground ground truth and prediction $G_{\text{fg}} := \bigvee_{\substack{c \in C \\ c \neq \text{bg}}} G_c$ and $P_{\text{fg}} := \bigvee_{\substack{c \in C \\ c \neq \text{bg}}} P_c$. Assuming mutually exclusive class segmentations we can compute

$$\text{DSC}_{\text{fg}} := \frac{2|P_{\text{fg}} \odot G_{\text{fg}}|}{|P_{\text{fg}}| + |G_{\text{fg}}|} = \frac{2|\overline{P_{\text{bg}}} \odot \overline{G_{\text{bg}}}|}{|\overline{P_{\text{bg}}}| + |\overline{G_{\text{bg}}}|}, \tag{3}$$

using only the background segmentation. In this equation, $\overline{P_{\text{bg}}}$ denotes the logic complement of the binary predication for the background class bg, and $\overline{G_{\text{bg}}}$ denotes the respective ground truth.

We employ cross-validation using five different data folds, each comprising of three scans for training, one for validation and one for testing. The validation set is used for adjusting the hyperparameters and monitoring convergence. Within every cross-evaluation fold, we use a different scan for the final testing.

4 Results and Discussion

To evaluate the contributions of different patch sizes, sampling strategies, data augmentation schemes and loss functions, we present quantitative results in Table 1. We investigate input patch sizes of 96, 128, and 160 px per dimension, chosen through balanced sampling **bal** or uniform sampling **unif**. The subscript **xent** stands for the use of the cross-entropy loss function alone instead of the full loss (1). With **d** we denote data augmentation with elastic deformations.

Not least because of the small data set available, there is considerable variance within the DSC scores of a given model and bone group, which impedes direct comparison of different models. No single model outperforms all others, although bigger patch sizes correspond to higher total scores. As for class imbalances, we note that the two models trained with a uniform sampler have the

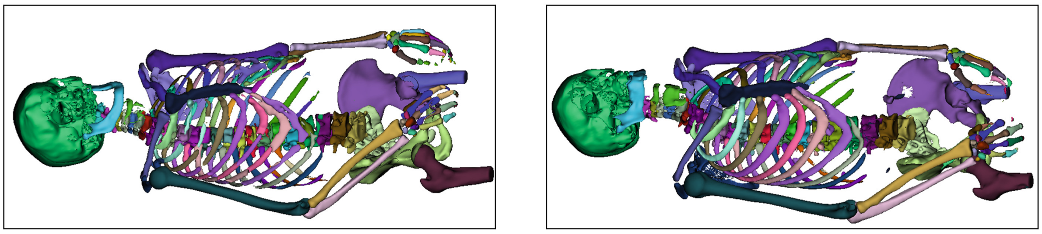


Fig. 3. Qualitative segmentation results created using $160_{\mathrm{bal,d}}$ and two exemplary CT scans for which no manual labels exist. The 3D views were created with 3D Slicer [6] and show an interpolated version of the data.

lowest detection ratio for bones in the hands. The balanced sampler thus seems to benefit the detection and segmentation of tiny bones. We indicate the time needed for one iteration of training. To ensure a fair comparison, we averaged 100 iterations trained on the same machine under stable conditions. Patch sizes profoundly influence the time needed per iteration. The resulting times range from close to 1 second for a patch of size 96^3 up to almost 9 seconds for patches sized 160^3. The loss function also influences the training time considerably, with pure cross-entropy taking only half as long as the combined loss function.

Because many of our limitations in computational resources stem from the combination of a 3D network with a large number of classes, we additionally provide the results obtained using a 2D U-Net. We trained this network as specified in [13] who used it successfully for non-distinct bone segmentation of whole-body CT images. This network leads to good results for the 2-class case (2D U-Net_{2c}), but it does not scale well to bone distinction, as our results of a 2D U-Net_{126c} – trained on the primary task – suggest.

A comparison with existing methods is made in Table 2. Since code and data sets are not publicly available, we compare the published results for different bones with our own. While the atlas method presented in [7] exhibits the best segmentation performance, their inference takes 20 min. They also require manual intervention if used on CT images that show only parts of an upper body. The two-step neural network presented in [17] was trained on 100 data sets and evaluated on 5. For the sacrum and L3, both our work and [17] show similar results. For bones that have a high chance of being confused with the ones below and above, their use of a shape model for landmark labelling and post-processing helps to keep scores for ribs and vertebrae high. It is, however, not clear how easily their approach could be adapted to accommodate for the segmentation of further bones, e.g. hands.

Qualitative results using two scans of unlabelled data are depicted in Fig. 3.

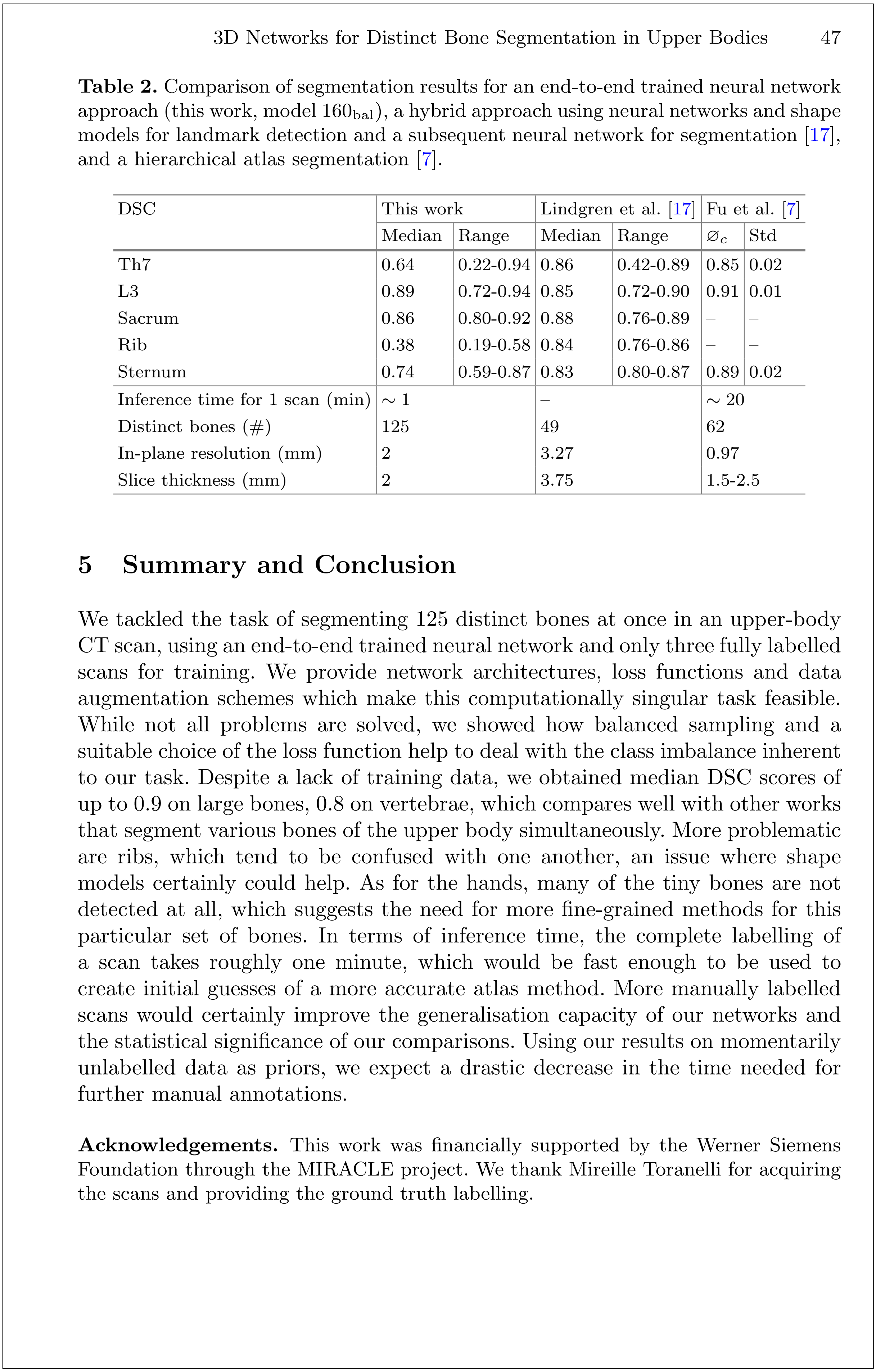


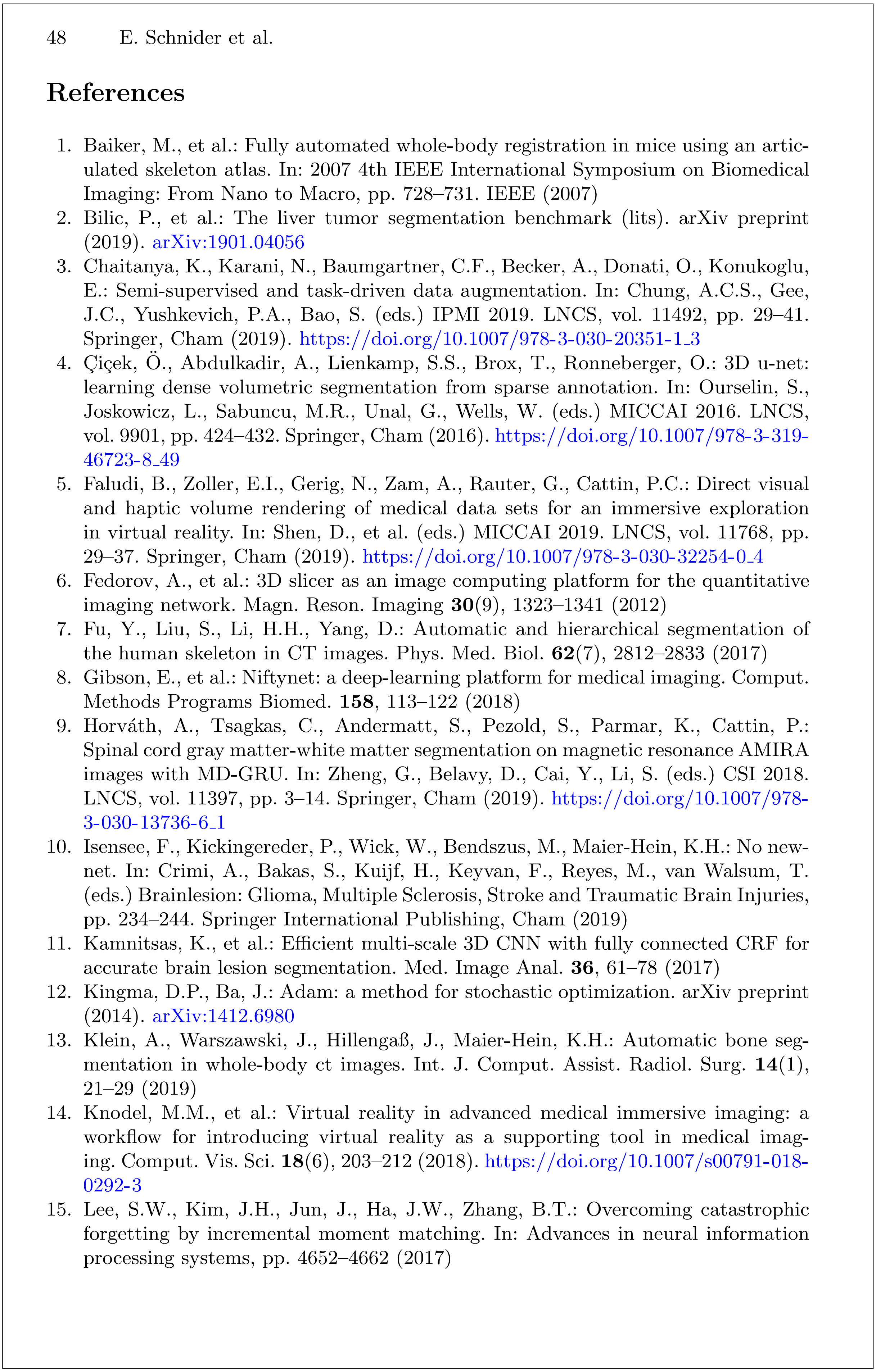
Table 2. Comparison of segmentation results for an end-to-end trained neural network approach (this work, model 160_{bal}), a hybrid approach using neural networks and shape models for landmark detection and a subsequent neural network for segmentation [17], and a hierarchical atlas segmentation [7].

DSC	This work		Lindgren et al. [17]		Fu et al. [7]	
	Median	Range	Median	Range	$\varnothing_c$	Std
Th7	0.64	0.22-0.94	0.86	0.42-0.89	0.85	0.02
L3	0.89	0.72-0.94	0.85	0.72-0.90	0.91	0.01
Sacrum	0.86	0.80-0.92	0.88	0.76-0.89	–	–
Rib	0.38	0.19-0.58	0.84	0.76-0.86	–	–
Sternum	0.74	0.59-0.87	0.83	0.80-0.87	0.89	0.02
Inference time for 1 scan (min)	∼ 1		–		∼ 20	
Distinct bones (#)	125		49		62	
In-plane resolution (mm)	2		3.27		0.97	
Slice thickness (mm)	2		3.75		1.5-2.5	

5 Summary and Conclusion

We tackled the task of segmenting 125 distinct bones at once in an upper-body CT scan, using an end-to-end trained neural network and only three fully labelled scans for training. We provide network architectures, loss functions and data augmentation schemes which make this computationally singular task feasible. While not all problems are solved, we showed how balanced sampling and a suitable choice of the loss function help to deal with the class imbalance inherent to our task. Despite a lack of training data, we obtained median DSC scores of up to 0.9 on large bones, 0.8 on vertebrae, which compares well with other works that segment various bones of the upper body simultaneously. More problematic are ribs, which tend to be confused with one another, an issue where shape models certainly could help. As for the hands, many of the tiny bones are not detected at all, which suggests the need for more fine-grained methods for this particular set of bones. In terms of inference time, the complete labelling of a scan takes roughly one minute, which would be fast enough to be used to create initial guesses of a more accurate atlas method. More manually labelled scans would certainly improve the generalisation capacity of our networks and the statistical significance of our comparisons. Using our results on momentarily unlabelled data as priors, we expect a drastic decrease in the time needed for further manual annotations.

Acknowledgements. This work was financially supported by the Werner Siemens Foundation through the MIRACLE project. We thank Mireille Toranelli for acquiring the scans and providing the ground truth labelling.



References

1. Baiker, M., et al.: Fully automated whole-body registration in mice using an articulated skeleton atlas. In: 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 728–731. IEEE (2007)
2. Bilic, P., et al.: The liver tumor segmentation benchmark (lits). arXiv preprint (2019). arXiv:1901.04056
3. Chaitanya, K., Karani, N., Baumgartner, C.F., Becker, A., Donati, O., Konukoglu, E.: Semi-supervised and task-driven data augmentation. In: Chung, A.C.S., Gee, J.C., Yushkevich, P.A., Bao, S. (eds.) IPMI 2019. LNCS, vol. 11492, pp. 29–41. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20351-1_3
4. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D u-net: learning dense volumetric segmentation from sparse annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_49
5. Faludi, B., Zoller, E.I., Gerig, N., Zam, A., Rauter, G., Cattin, P.C.: Direct visual and haptic volume rendering of medical data sets for an immersive exploration in virtual reality. In: Shen, D., et al. (eds.) MICCAI 2019. LNCS, vol. 11768, pp. 29–37. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32254-0_4
6. Fedorov, A., et al.: 3D slicer as an image computing platform for the quantitative imaging network. Magn. Reson. Imaging **30**(9), 1323–1341 (2012)
7. Fu, Y., Liu, S., Li, H.H., Yang, D.: Automatic and hierarchical segmentation of the human skeleton in CT images. Phys. Med. Biol. **62**(7), 2812–2833 (2017)
8. Gibson, E., et al.: Niftynet: a deep-learning platform for medical imaging. Comput. Methods Programs Biomed. **158**, 113–122 (2018)
9. Horváth, A., Tsagkas, C., Andermatt, S., Pezold, S., Parmar, K., Cattin, P.: Spinal cord gray matter-white matter segmentation on magnetic resonance AMIRA images with MD-GRU. In: Zheng, G., Belavy, D., Cai, Y., Li, S. (eds.) CSI 2018. LNCS, vol. 11397, pp. 3–14. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-13736-6_1
10. Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., Maier-Hein, K.H.: No new-net. In: Crimi, A., Bakas, S., Kuijf, H., Keyvan, F., Reyes, M., van Walsum, T. (eds.) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 234–244. Springer International Publishing, Cham (2019)
11. Kamnitsas, K., et al.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med. Image Anal. **36**, 61–78 (2017)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint (2014). arXiv:1412.6980
13. Klein, A., Warszawski, J., Hillengaß, J., Maier-Hein, K.H.: Automatic bone segmentation in whole-body ct images. Int. J. Comput. Assist. Radiol. Surg. **14**(1), 21–29 (2019)
14. Knodel, M.M., et al.: Virtual reality in advanced medical immersive imaging: a workflow for introducing virtual reality as a supporting tool in medical imaging. Comput. Vis. Sci. **18**(6), 203–212 (2018). https://doi.org/10.1007/s00791-018-0292-3
15. Lee, S.W., Kim, J.H., Jun, J., Ha, J.W., Zhang, B.T.: Overcoming catastrophic forgetting by incremental moment matching. In: Advances in neural information processing systems, pp. 4652–4662 (2017)

16. Lessmann, N., van Ginneken, B., de Jong, P.A., Išgum, I.: Iterative fully convolutional neural networks for automatic vertebra segmentation and identification. Med. Image Anal. **53**, 142–155 (2019)
17. Lindgren Belal, S., et al.: Deep learning for segmentation of 49 selected bones in CT scans: first step in automated PET/CT-based 3D quantification of skeletal metastases. Eur. J. Radiol. **113**, 89–95 (2019)
18. Pérez-Carrasco, J.A., Acha, B., Suárez-Mejías, C., López-Guerra, J.L., Serrano, C.: Joint segmentation of bones and muscles using an intensity and histogram-based energy minimization approach. Comput. Methods Programs Biomed. **156**, 85–95 (2018)
19. Rahbani, D., Morel-Forster, A., Madsen, D., Lüthi, M., Vetter, T.: Robust registration of statistical shape models for unsupervised pathology annotation. In: Zhou, L., et al. (eds.) LABELS/HAL-MICCAI/CuRIOUS -2019. LNCS, vol. 11851, pp. 13–21. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33642-4_2
20. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
21. Sarkalkan, N., Weinans, H., Zadpoor, A.A.: Statistical shape and appearance models of bones. Bone **60**, 129–140 (2014)
22. Seim, H., Kainmueller, D., Heller, M., Lamecker, H., Zachow, S., Hege, H.C.: Automatic segmentation of the pelvic bones from ct data based on a statistical shape model. VCBM **8**, 93–100 (2008)
23. Sekuboyina, A., et al.: Verse: a vertebrae labelling and segmentation benchmark. arXiv preprint (2020). arXiv:2001.09193
24. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: beyond empirical risk minimization. arXiv preprint (2017). arXiv:1710.09412

Chapter 6

Ensemble Uncertainty as a Criterion for Dataset Expansion in Distinct Bone Segmentation from Upper-Body CT Images.

We expanded our data set by leveraging different ensemble approaches and creating as-good-as possible automated segmentations on unannotated data. In cooperation with the university's anatomical department, we corrected those ensemble predictions and turned them into new training data. We then evaluated how useful this new training data was to obtain a more accurate model. We also compared the usefulness of the new data against the data's ensemble uncertainty. Against our expectations, the ensemble uncertainty of newly labelled data was no predictor of its usefulness. It was, however, a predictor of the quality of the automated segmentation and thus gave a measure of how much manual correction work was necessary to turn a given scan into new training data.

Technical Report. The following manuscript was presented orally at the conference for *Computer Assisted Radiology and Surgery* (CARS), June 2022, Tokyo, Japan. The report [113] has been submitted to arxiv.org[1].

[1] https://arxiv.org/abs/2208.09216

Ensemble uncertainty as a criterion for dataset expansion in distinct bone segmentation from upper-body CT images

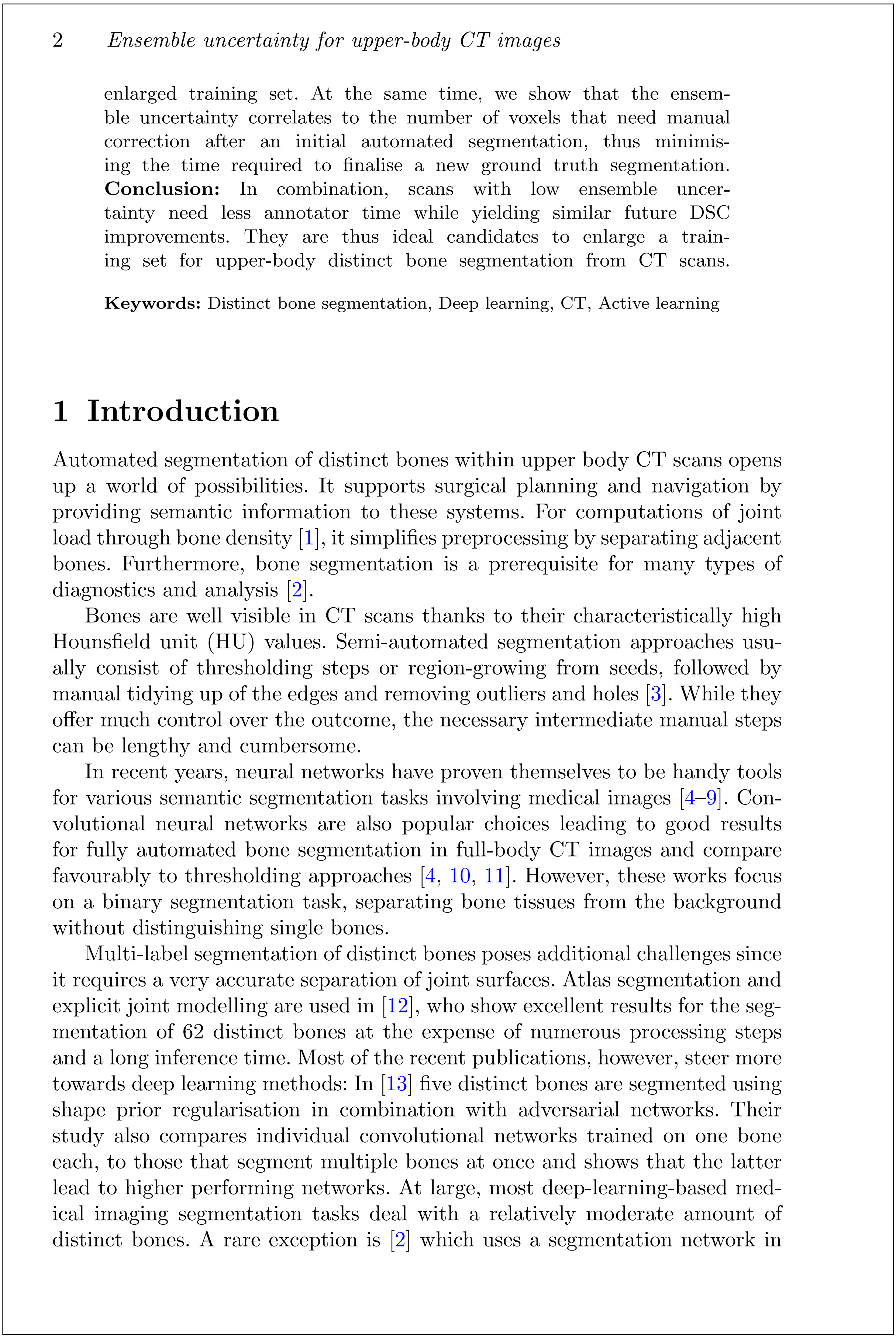
Eva Schnider[1*], Antal Huck[1], Mireille Toranelli[2], Georg Rauter[1], Azhar Zam[1], Magdalena Müller-Gerbl[2] and Philippe Cattin[1]

[1*]Department of Biomedical Engineering, University of Basel, Gewerbestrasse 14, Allschwil, 4123 Switzerland.
[2]Department of Biomedicine, University of Basel, Basel, Switzerland.

*Corresponding author(s). E-mail(s): eva.schnider@unibas.ch;

Abstract

Purpose: The localisation and segmentation of individual bones is an important preprocessing step in many planning and navigation applications. It is, however, a time-consuming and repetitive task if done manually. This is true not only for clinical practice but also for the acquisition of training data. We therefore not only present an end-to-end learnt algorithm that is capable of segmenting 125 distinct bones in an upper-body CT, but also provide an ensemble-based uncertainty measure that helps to single out scans to enlarge the training dataset with.
Methods: We create fully automated end-to-end learnt segmentations using a neural network architecture inspired by the 3D-Unet and fully supervised training. The results are improved using ensembles and inference-time augmentation. We examine the relationship of ensemble-uncertainty to an unlabelled scan's prospective usefulness as part of the training dataset.
Results: Our methods are evaluated on an in-house dataset of 16 upper-body CT scans with a resolution of 2 mm per dimension. Taking into account all 125 bones in our label set, our most successful ensemble achieves a median dice score coefficient of 0.83. We find a lack of correlation between a scan's ensemble uncertainty and its prospective influence on the accuracies achieved within an



enlarged training set. At the same time, we show that the ensemble uncertainty correlates to the number of voxels that need manual correction after an initial automated segmentation, thus minimising the time required to finalise a new ground truth segmentation. **Conclusion:** In combination, scans with low ensemble uncertainty need less annotator time while yielding similar future DSC improvements. They are thus ideal candidates to enlarge a training set for upper-body distinct bone segmentation from CT scans.

Keywords: Distinct bone segmentation, Deep learning, CT, Active learning

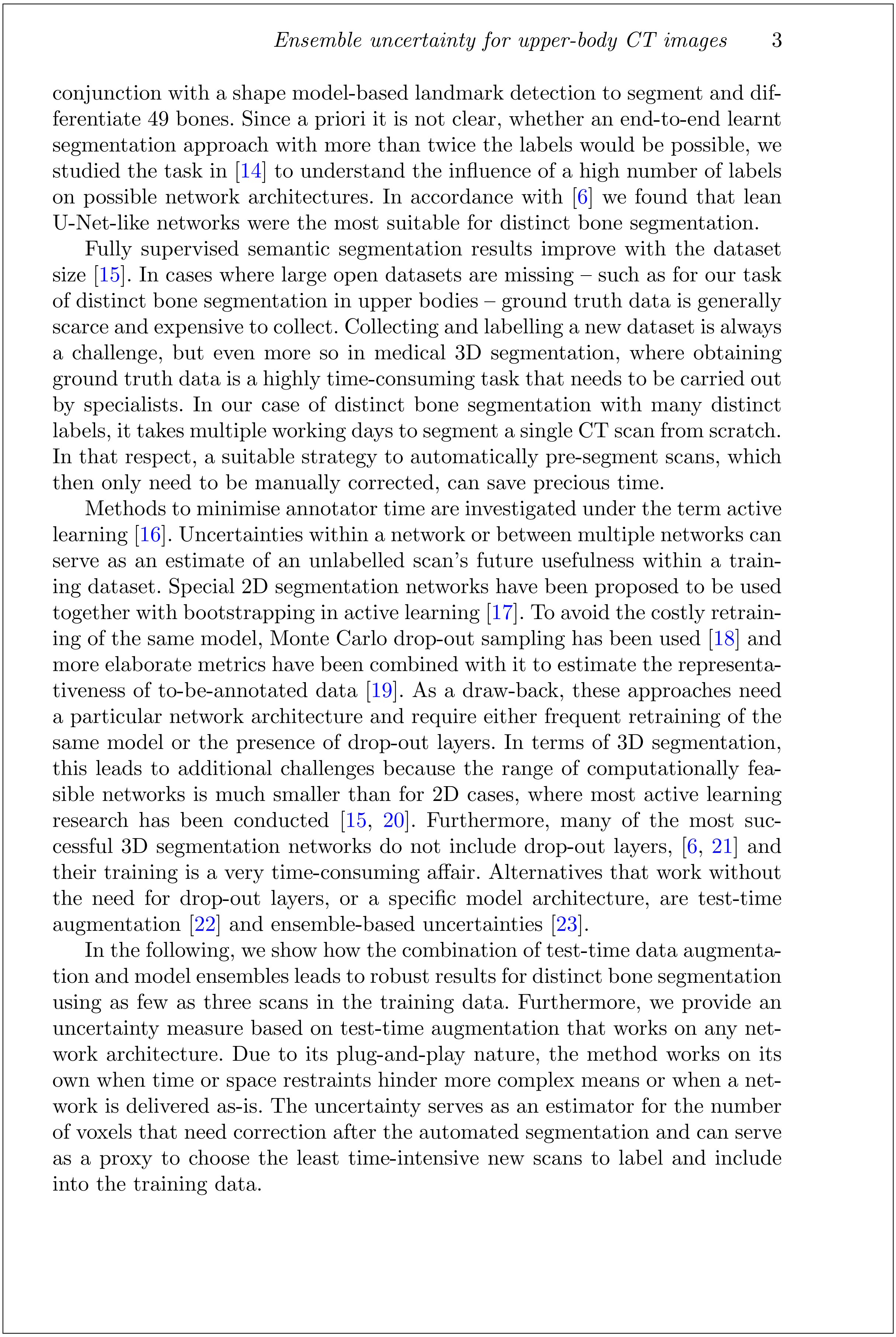
1 Introduction

Automated segmentation of distinct bones within upper body CT scans opens up a world of possibilities. It supports surgical planning and navigation by providing semantic information to these systems. For computations of joint load through bone density [1], it simplifies preprocessing by separating adjacent bones. Furthermore, bone segmentation is a prerequisite for many types of diagnostics and analysis [2].

Bones are well visible in CT scans thanks to their characteristically high Hounsfield unit (HU) values. Semi-automated segmentation approaches usually consist of thresholding steps or region-growing from seeds, followed by manual tidying up of the edges and removing outliers and holes [3]. While they offer much control over the outcome, the necessary intermediate manual steps can be lengthy and cumbersome.

In recent years, neural networks have proven themselves to be handy tools for various semantic segmentation tasks involving medical images [4–9]. Convolutional neural networks are also popular choices leading to good results for fully automated bone segmentation in full-body CT images and compare favourably to thresholding approaches [4, 10, 11]. However, these works focus on a binary segmentation task, separating bone tissues from the background without distinguishing single bones.

Multi-label segmentation of distinct bones poses additional challenges since it requires a very accurate separation of joint surfaces. Atlas segmentation and explicit joint modelling are used in [12], who show excellent results for the segmentation of 62 distinct bones at the expense of numerous processing steps and a long inference time. Most of the recent publications, however, steer more towards deep learning methods: In [13] five distinct bones are segmented using shape prior regularisation in combination with adversarial networks. Their study also compares individual convolutional networks trained on one bone each, to those that segment multiple bones at once and shows that the latter lead to higher performing networks. At large, most deep-learning-based medical imaging segmentation tasks deal with a relatively moderate amount of distinct bones. A rare exception is [2] which uses a segmentation network in

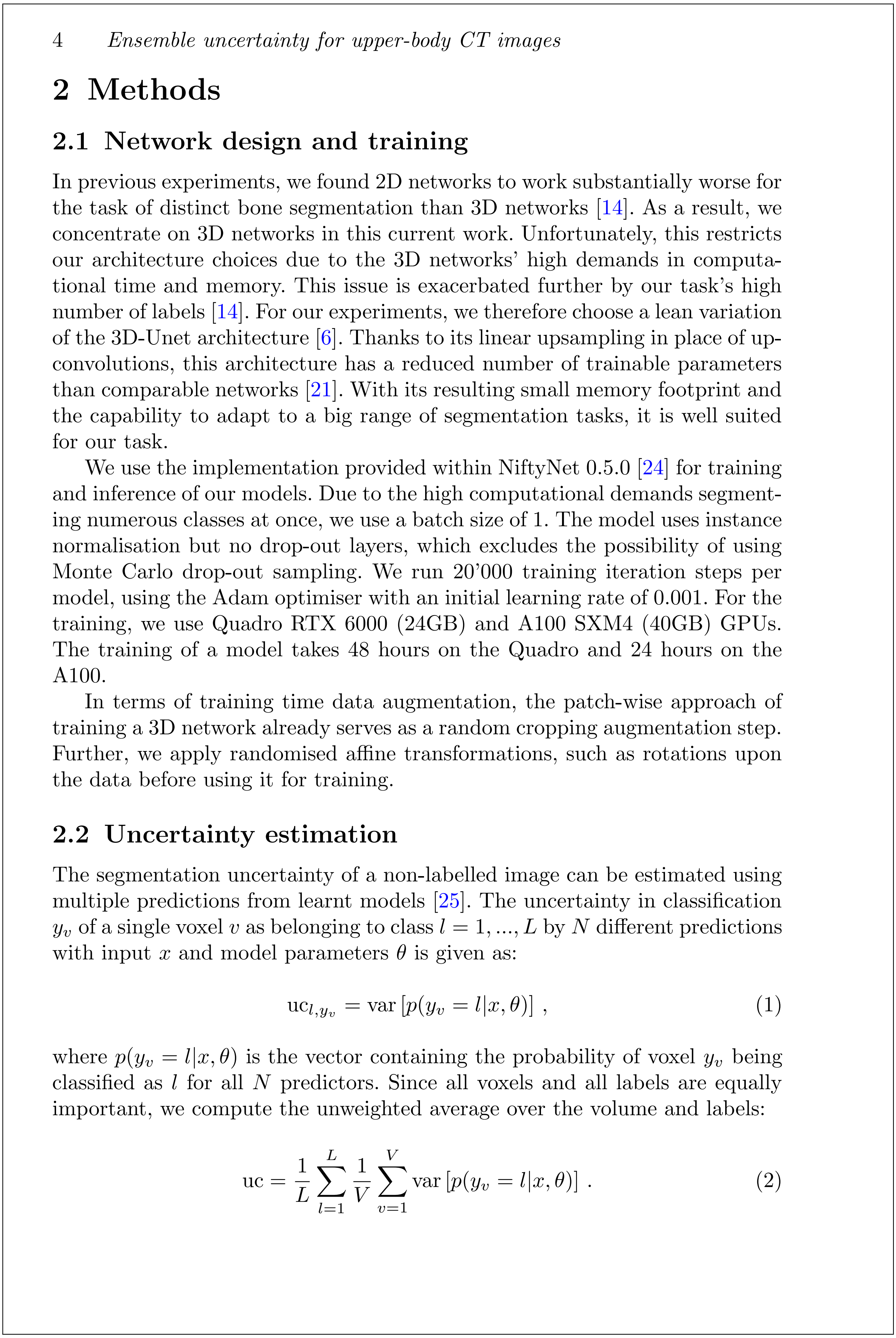


conjunction with a shape model-based landmark detection to segment and differentiate 49 bones. Since a priori it is not clear, whether an end-to-end learnt segmentation approach with more than twice the labels would be possible, we studied the task in [14] to understand the influence of a high number of labels on possible network architectures. In accordance with [6] we found that lean U-Net-like networks were the most suitable for distinct bone segmentation.

Fully supervised semantic segmentation results improve with the dataset size [15]. In cases where large open datasets are missing – such as for our task of distinct bone segmentation in upper bodies – ground truth data is generally scarce and expensive to collect. Collecting and labelling a new dataset is always a challenge, but even more so in medical 3D segmentation, where obtaining ground truth data is a highly time-consuming task that needs to be carried out by specialists. In our case of distinct bone segmentation with many distinct labels, it takes multiple working days to segment a single CT scan from scratch. In that respect, a suitable strategy to automatically pre-segment scans, which then only need to be manually corrected, can save precious time.

Methods to minimise annotator time are investigated under the term active learning [16]. Uncertainties within a network or between multiple networks can serve as an estimate of an unlabelled scan's future usefulness within a training dataset. Special 2D segmentation networks have been proposed to be used together with bootstrapping in active learning [17]. To avoid the costly retraining of the same model, Monte Carlo drop-out sampling has been used [18] and more elaborate metrics have been combined with it to estimate the representativeness of to-be-annotated data [19]. As a draw-back, these approaches need a particular network architecture and require either frequent retraining of the same model or the presence of drop-out layers. In terms of 3D segmentation, this leads to additional challenges because the range of computationally feasible networks is much smaller than for 2D cases, where most active learning research has been conducted [15, 20]. Furthermore, many of the most successful 3D segmentation networks do not include drop-out layers, [6, 21] and their training is a very time-consuming affair. Alternatives that work without the need for drop-out layers, or a specific model architecture, are test-time augmentation [22] and ensemble-based uncertainties [23].

In the following, we show how the combination of test-time data augmentation and model ensembles leads to robust results for distinct bone segmentation using as few as three scans in the training data. Furthermore, we provide an uncertainty measure based on test-time augmentation that works on any network architecture. Due to its plug-and-play nature, the method works on its own when time or space restraints hinder more complex means or when a network is delivered as-is. The uncertainty serves as an estimator for the number of voxels that need correction after the automated segmentation and can serve as a proxy to choose the least time-intensive new scans to label and include into the training data.



2 Methods

2.1 Network design and training

In previous experiments, we found 2D networks to work substantially worse for the task of distinct bone segmentation than 3D networks [14]. As a result, we concentrate on 3D networks in this current work. Unfortunately, this restricts our architecture choices due to the 3D networks' high demands in computational time and memory. This issue is exacerbated further by our task's high number of labels [14]. For our experiments, we therefore choose a lean variation of the 3D-Unet architecture [6]. Thanks to its linear upsampling in place of up-convolutions, this architecture has a reduced number of trainable parameters than comparable networks [21]. With its resulting small memory footprint and the capability to adapt to a big range of segmentation tasks, it is well suited for our task.

We use the implementation provided within NiftyNet 0.5.0 [24] for training and inference of our models. Due to the high computational demands segmenting numerous classes at once, we use a batch size of 1. The model uses instance normalisation but no drop-out layers, which excludes the possibility of using Monte Carlo drop-out sampling. We run 20'000 training iteration steps per model, using the Adam optimiser with an initial learning rate of 0.001. For the training, we use Quadro RTX 6000 (24GB) and A100 SXM4 (40GB) GPUs. The training of a model takes 48 hours on the Quadro and 24 hours on the A100.

In terms of training time data augmentation, the patch-wise approach of training a 3D network already serves as a random cropping augmentation step. Further, we apply randomised affine transformations, such as rotations upon the data before using it for training.

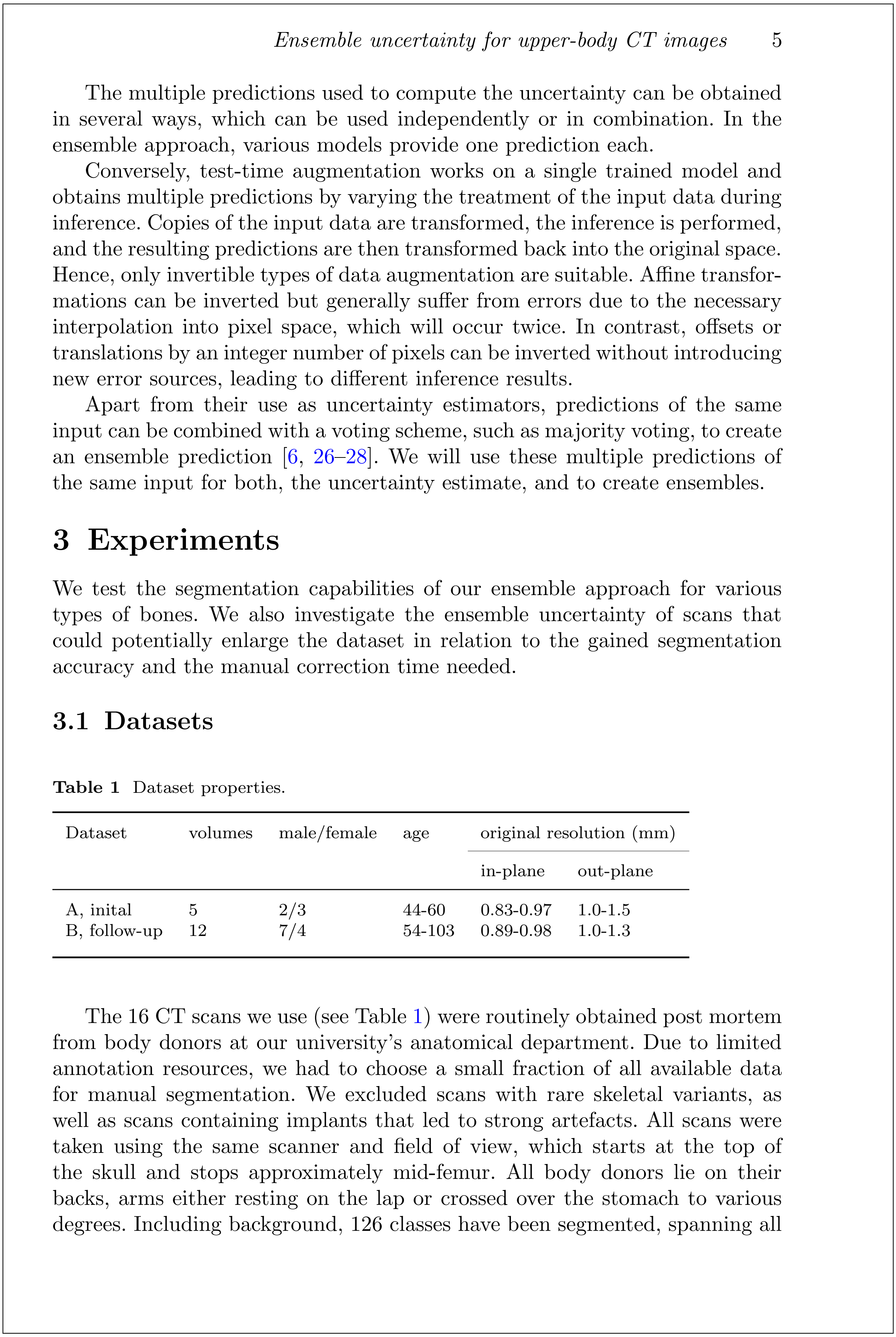
2.2 Uncertainty estimation

The segmentation uncertainty of a non-labelled image can be estimated using multiple predictions from learnt models [25]. The uncertainty in classification y_v of a single voxel v as belonging to class $l = 1, ..., L$ by N different predictions with input x and model parameters θ is given as:

$$\mathrm{uc}_{l,y_v} = \mathrm{var}\left[p(y_v = l|x, \theta)\right] \ , \qquad (1)$$

where $p(y_v = l|x, \theta)$ is the vector containing the probability of voxel y_v being classified as l for all N predictors. Since all voxels and all labels are equally important, we compute the unweighted average over the volume and labels:

$$\mathrm{uc} = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{V}\sum_{v=1}^{V}\mathrm{var}\left[p(y_v = l|x, \theta)\right] \ . \qquad (2)$$



The multiple predictions used to compute the uncertainty can be obtained in several ways, which can be used independently or in combination. In the ensemble approach, various models provide one prediction each.

Conversely, test-time augmentation works on a single trained model and obtains multiple predictions by varying the treatment of the input data during inference. Copies of the input data are transformed, the inference is performed, and the resulting predictions are then transformed back into the original space. Hence, only invertible types of data augmentation are suitable. Affine transformations can be inverted but generally suffer from errors due to the necessary interpolation into pixel space, which will occur twice. In contrast, offsets or translations by an integer number of pixels can be inverted without introducing new error sources, leading to different inference results.

Apart from their use as uncertainty estimators, predictions of the same input can be combined with a voting scheme, such as majority voting, to create an ensemble prediction [6, 26–28]. We will use these multiple predictions of the same input for both, the uncertainty estimate, and to create ensembles.

3 Experiments

We test the segmentation capabilities of our ensemble approach for various types of bones. We also investigate the ensemble uncertainty of scans that could potentially enlarge the dataset in relation to the gained segmentation accuracy and the manual correction time needed.

3.1 Datasets

Table 1 Dataset properties.

Dataset	volumes	male/female	age	original resolution (mm)	
				in-plane	out-plane
A, inital	5	2/3	44-60	0.83-0.97	1.0-1.5
B, follow-up	12	7/4	54-103	0.89-0.98	1.0-1.3

The 16 CT scans we use (see Table 1) were routinely obtained post mortem from body donors at our university's anatomical department. Due to limited annotation resources, we had to choose a small fraction of all available data for manual segmentation. We excluded scans with rare skeletal variants, as well as scans containing implants that led to strong artefacts. All scans were taken using the same scanner and field of view, which starts at the top of the skull and stops approximately mid-femur. All body donors lie on their backs, arms either resting on the lap or crossed over the stomach to various degrees. Including background, 126 classes have been segmented, spanning all

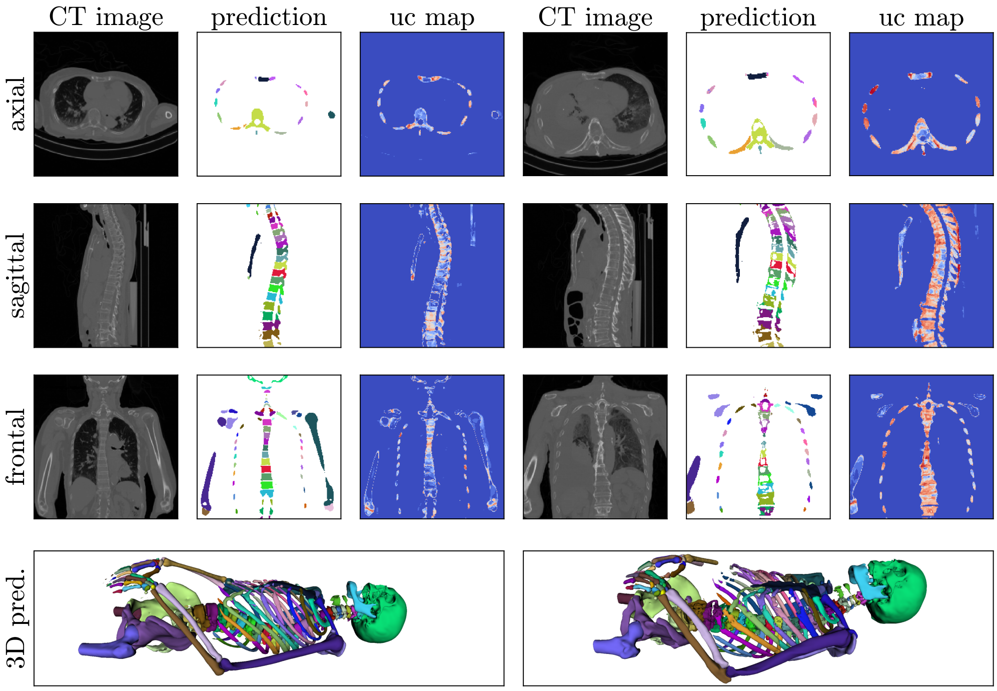


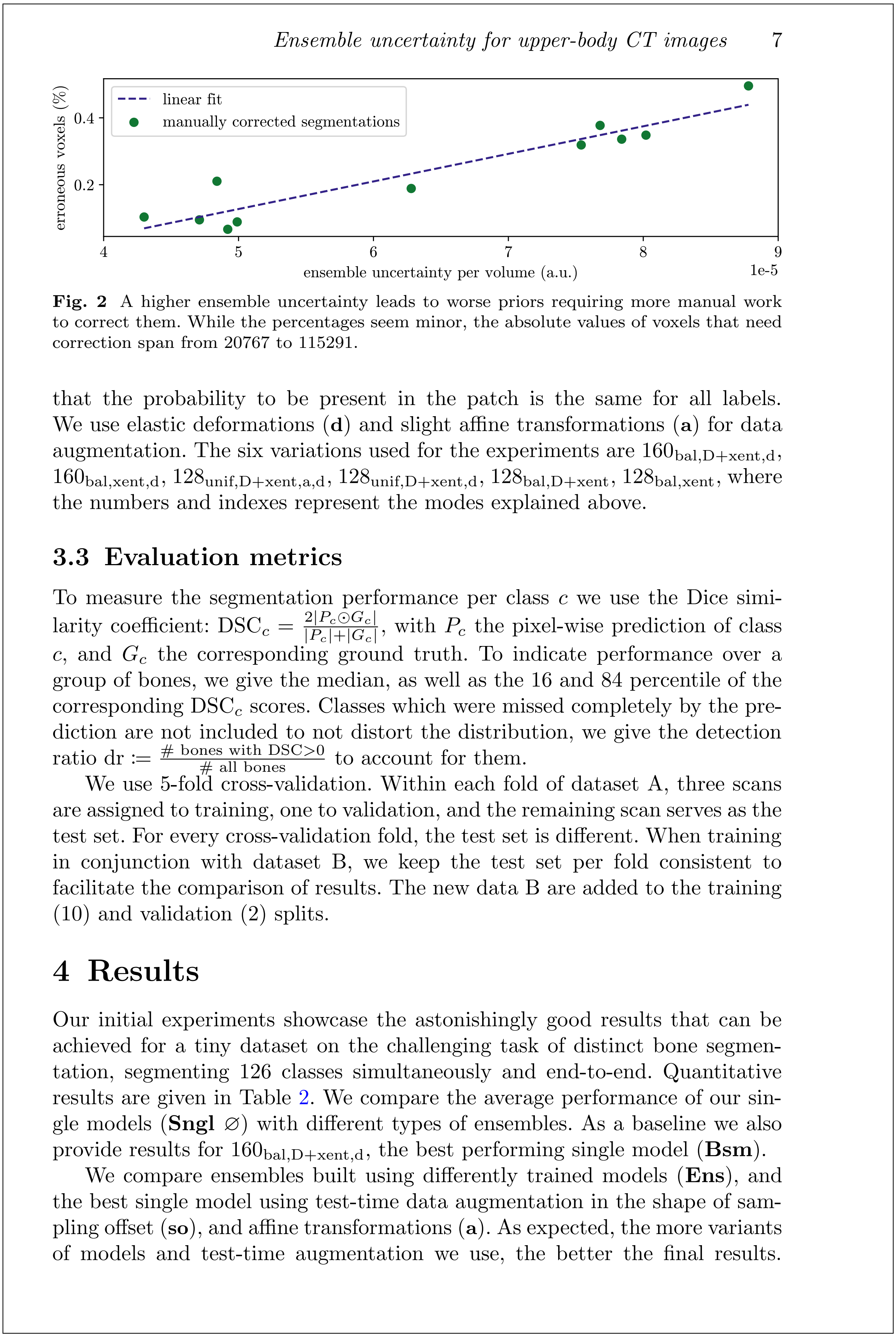
Fig. 1 Qualitative results were obtained using the ensemble approach of combining six model variations on a hold-out set of unlabelled scans. Shown are CT images predicted label map, and uncertainty map for both the scan with lowest (left) and highest (right) mean uncertainty (2.2).

upper body bones, pelvis, and femurs. the subjects' advanced age (see Table 1) manifests in different levels of scoliosis and calcifications.

Dataset A has already been used and described in [14]. The remaining eleven scans of dataset B were pre-segmented using our ensemble models, and the labelling afterwards manually improved. Nevertheless, the time needed to finish a manual segmentation still exceeded a working day per scan due to the many distinct classes. We re-sample our data to a uniform resolution of 2 mm in all three dimensions which leaves us with scans of $\sim 265 \times 256 \times 512$ voxels. The smaller resolution reduces I/O times during training tremendously and allows us to capture more body context within an input patch. The labels of the various bones are highly imbalanced, which complicates model training. While many bones in the hands only span a few dozen voxels, there are also big bones such as the skull, femur, and pelvis, which easily exceed 10'000 voxels.

3.2 Ensemble Model Variations

To obtain different predictors for our ensemble, we perform the training using a range hyperparameter settings that have been found to work well for the given task [14]. Input patch sizes range from 96 px - 160 px per dimension, the loss function used is either pure multi-label cross-entropy (**xent**) or a linear combination of the DSC loss [29] and the cross-entropy loss (**D+xent**). Patches are sampled uniformly random (**unif**) or in a balanced fashion (**bal**), such



linear fit
manually corrected segmentations
erroneous voxels (%)
0.4
0.2
4
5
6
7
8
9
ensemble uncertainty per volume (a.u.)
1e-5

Fig. 2 A higher ensemble uncertainty leads to worse priors requiring more manual work to correct them. While the percentages seem minor, the absolute values of voxels that need correction span from 20767 to 115291.

that the probability to be present in the patch is the same for all labels. We use elastic deformations (**d**) and slight affine transformations (**a**) for data augmentation. The six variations used for the experiments are $160_{\mathrm{bal,D+xent,d}}$, $160_{\mathrm{bal,xent,d}}$, $128_{\mathrm{unif,D+xent,a,d}}$, $128_{\mathrm{unif,D+xent,d}}$, $128_{\mathrm{bal,D+xent}}$, $128_{\mathrm{bal,xent}}$, where the numbers and indexes represent the modes explained above.

3.3 Evaluation metrics

To measure the segmentation performance per class c we use the Dice similarity coefficient: $\mathrm{DSC}_c = \frac{2|P_c \odot G_c|}{|P_c|+|G_c|}$, with P_c the pixel-wise prediction of class c, and G_c the corresponding ground truth. To indicate performance over a group of bones, we give the median, as well as the 16 and 84 percentile of the corresponding DSC_c scores. Classes which were missed completely by the prediction are not included to not distort the distribution, we give the detection ratio $\mathrm{dr} := \frac{\#\text{ bones with DSC}>0}{\#\text{ all bones}}$ to account for them.

We use 5-fold cross-validation. Within each fold of dataset A, three scans are assigned to training, one to validation, and the remaining scan serves as the test set. For every cross-validation fold, the test set is different. When training in conjunction with dataset B, we keep the test set per fold consistent to facilitate the comparison of results. The new data B are added to the training (10) and validation (2) splits.

4 Results

Our initial experiments showcase the astonishingly good results that can be achieved for a tiny dataset on the challenging task of distinct bone segmentation, segmenting 126 classes simultaneously and end-to-end. Quantitative results are given in Table 2. We compare the average performance of our single models (**Sngl** $\varnothing$) with different types of ensembles. As a baseline we also provide results for $160_{\mathrm{bal,D+xent,d}}$, the best performing single model (**Bsm**).

We compare ensembles built using differently trained models (**Ens**), and the best single model using test-time data augmentation in the shape of sampling offset (**so**), and affine transformations (**a**). As expected, the more variants of models and test-time augmentation we use, the better the final results.

Table 2 Ablation results using single models and various ensemble types on dataset A. For comparison we also give results on the final full dataset A+B.

	Vertebrae	Ribs	Hands	Large	All	es[1]	ds[2]
Sngl∅	$0.81^{+0.09}_{-0.24}$	$0.58^{+0.22}_{-0.28}$	$0.54^{+0.26}_{-0.41}$(52%)	$0.88^{+0.05}_{-0.10}$	$0.72^{+0.17}_{-0.40}$(78%)	1	3
Bsm	$0.82^{+0.09}_{-0.17}$	$0.58^{+0.21}_{-0.27}$	$0.67^{+0.18}_{-0.39}$(58%)	$0.88^{+0.04}_{-0.11}$	$0.75^{+0.14}_{-0.38}$(80%)	1	3
Bsm$_{\text{so}}$	$0.86^{+0.06}_{-0.19}$	$0.65^{+0.19}_{-0.33}$	$0.58^{+0.24}_{-0.38}$(58%)	$0.89^{+0.04}_{-0.07}$	$0.76^{+0.14}_{-0.38}$(81%)	7	3
Bsm$_{\text{a+so}}$	$0.85^{+0.07}_{-0.17}$	$0.60^{+0.21}_{-0.31}$	$0.56^{+0.22}_{-0.39}$(54%)	$0.89^{+0.04}_{-0.08}$	$0.75^{+0.14}_{-0.40}$(79%)	7	3
Ens	$0.87^{+0.06}_{-0.18}$	$0.66^{+0.20}_{-0.33}$	$0.65^{+0.19}_{-0.50}$(57%)	$\mathbf{0.93}^{+\mathbf{0.03}}_{-\mathbf{0.03}}$	$0.80^{+0.12}_{-0.39}$(80%)	6	3
Ens$_{\text{so}}$	$0.88^{+0.06}_{-0.18}$	$0.68^{+0.19}_{-0.31}$	$0.62^{+0.24}_{-0.48}$(52%)	$0.92^{+0.03}_{-0.03}$	$0.83^{+0.10}_{-0.41}$(78%)	42	3
A+B	$\mathbf{0.89}^{+\mathbf{0.04}}_{-\mathbf{0.15}}$	$\mathbf{0.80}^{+\mathbf{0.07}}_{-\mathbf{0.15}}$	$\mathbf{0.79}^{+\mathbf{0.09}}_{-\mathbf{0.37}}$**(69%)**	$0.91^{+0.04}_{-0.08}$	$\mathbf{0.83}^{+\mathbf{0.08}}_{-\mathbf{0.22}}$**(85%)**	1	13

Results in DSC with the detection rate dr in brackets if it is less than 100%.

[1] The number of inferences used to compute the results is indicated as ensemble size es.

[2] The number of images in the training split is indicated as dataset size ds.

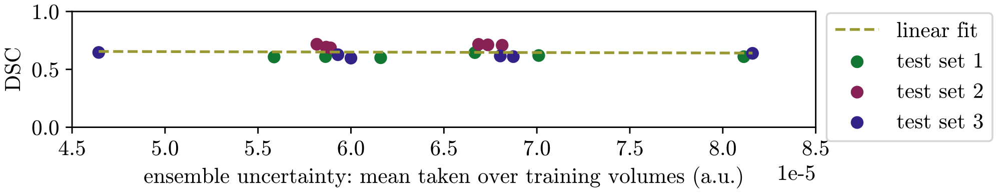


Fig. 3 Comparison of summed mean uncertainty of the training set and the resulting Dice score. We use three different test sets combined with 6 training sets each.

A depiction of qualitative segmentation results and uncertainty maps can be found in Figure 1.

For the follow-up experiments, we created a ground truth labelling for dataset B by using the hitherto best ensemble prediction Ens$_{\text{so}}$ to create a segmentation that was then cleaned up manually. We provide qualitative segmentation results achieved when training the initial dataset A alongside dataset B as a comparison to our ensemble results. Using more data – training with 13 instead of 3 scans – naturally improves results. The improvement is particularly evident for hands and ribs. On vertebrae and large bones, the difference is much smaller and the ensemble performs almost as good as the model trained on both A and B.

To analyze the use of the uncertainty metric, we defined three test sets, and for each of those then six training sets consisting of dataset A plus three scans from dataset B. The uncertainty of the scans in B had been established using the ensemble Ens$_{\text{so}}$. We trained the resulting 18 cases and plot the mean uncertainty of the training set against the DSC achieved on the test set (see

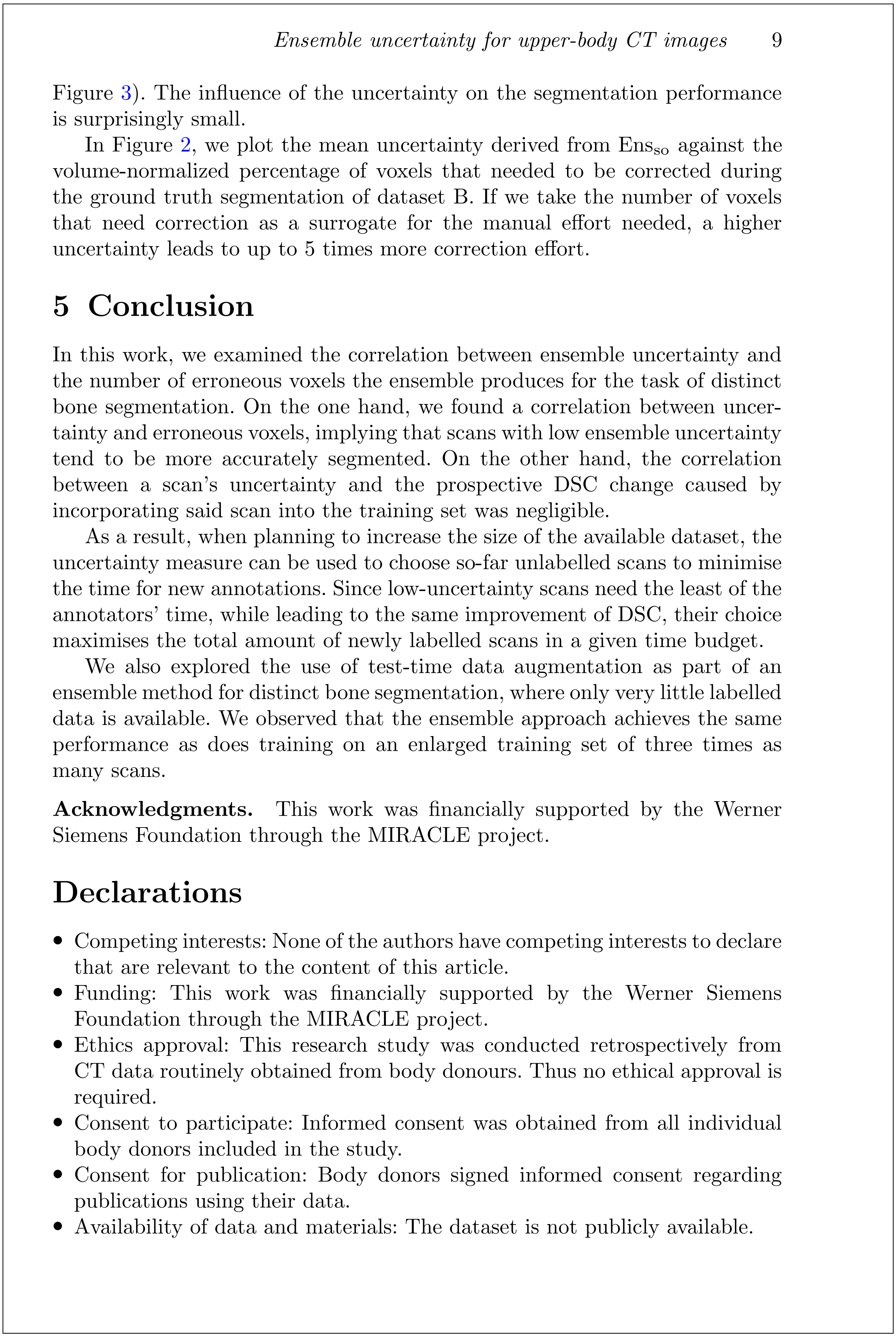


Figure 3). The influence of the uncertainty on the segmentation performance is surprisingly small.

In Figure 2, we plot the mean uncertainty derived from Ens_{so} against the volume-normalized percentage of voxels that needed to be corrected during the ground truth segmentation of dataset B. If we take the number of voxels that need correction as a surrogate for the manual effort needed, a higher uncertainty leads to up to 5 times more correction effort.

5 Conclusion

In this work, we examined the correlation between ensemble uncertainty and the number of erroneous voxels the ensemble produces for the task of distinct bone segmentation. On the one hand, we found a correlation between uncertainty and erroneous voxels, implying that scans with low ensemble uncertainty tend to be more accurately segmented. On the other hand, the correlation between a scan's uncertainty and the prospective DSC change caused by incorporating said scan into the training set was negligible.

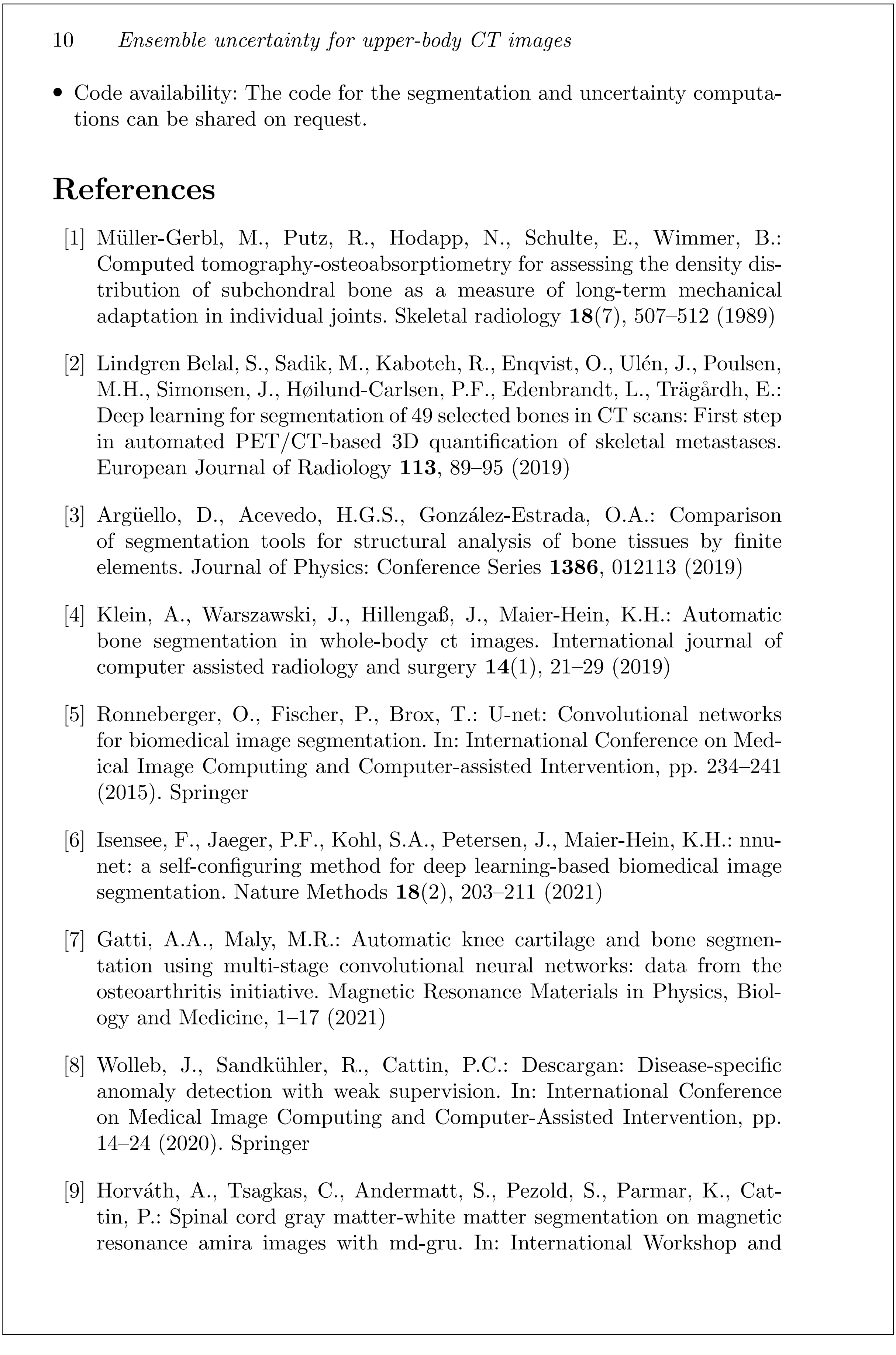
As a result, when planning to increase the size of the available dataset, the uncertainty measure can be used to choose so-far unlabelled scans to minimise the time for new annotations. Since low-uncertainty scans need the least of the annotators' time, while leading to the same improvement of DSC, their choice maximises the total amount of newly labelled scans in a given time budget.

We also explored the use of test-time data augmentation as part of an ensemble method for distinct bone segmentation, where only very little labelled data is available. We observed that the ensemble approach achieves the same performance as does training on an enlarged training set of three times as many scans.

Acknowledgments. This work was financially supported by the Werner Siemens Foundation through the MIRACLE project.

Declarations

- Competing interests: None of the authors have competing interests to declare that are relevant to the content of this article.
- Funding: This work was financially supported by the Werner Siemens Foundation through the MIRACLE project.
- Ethics approval: This research study was conducted retrospectively from CT data routinely obtained from body donours. Thus no ethical approval is required.
- Consent to participate: Informed consent was obtained from all individual body donors included in the study.
- Consent for publication: Body donors signed informed consent regarding publications using their data.
- Availability of data and materials: The dataset is not publicly available.



- Code availability: The code for the segmentation and uncertainty computations can be shared on request.

References

[1] Müller-Gerbl, M., Putz, R., Hodapp, N., Schulte, E., Wimmer, B.: Computed tomography-osteoabsorptiometry for assessing the density distribution of subchondral bone as a measure of long-term mechanical adaptation in individual joints. Skeletal radiology **18**(7), 507–512 (1989)

[2] Lindgren Belal, S., Sadik, M., Kaboteh, R., Enqvist, O., Ulén, J., Poulsen, M.H., Simonsen, J., Høilund-Carlsen, P.F., Edenbrandt, L., Trägårdh, E.: Deep learning for segmentation of 49 selected bones in CT scans: First step in automated PET/CT-based 3D quantification of skeletal metastases. European Journal of Radiology **113**, 89–95 (2019)

[3] Argüello, D., Acevedo, H.G.S., González-Estrada, O.A.: Comparison of segmentation tools for structural analysis of bone tissues by finite elements. Journal of Physics: Conference Series **1386**, 012113 (2019)

[4] Klein, A., Warszawski, J., Hillengaß, J., Maier-Hein, K.H.: Automatic bone segmentation in whole-body ct images. International journal of computer assisted radiology and surgery **14**(1), 21–29 (2019)

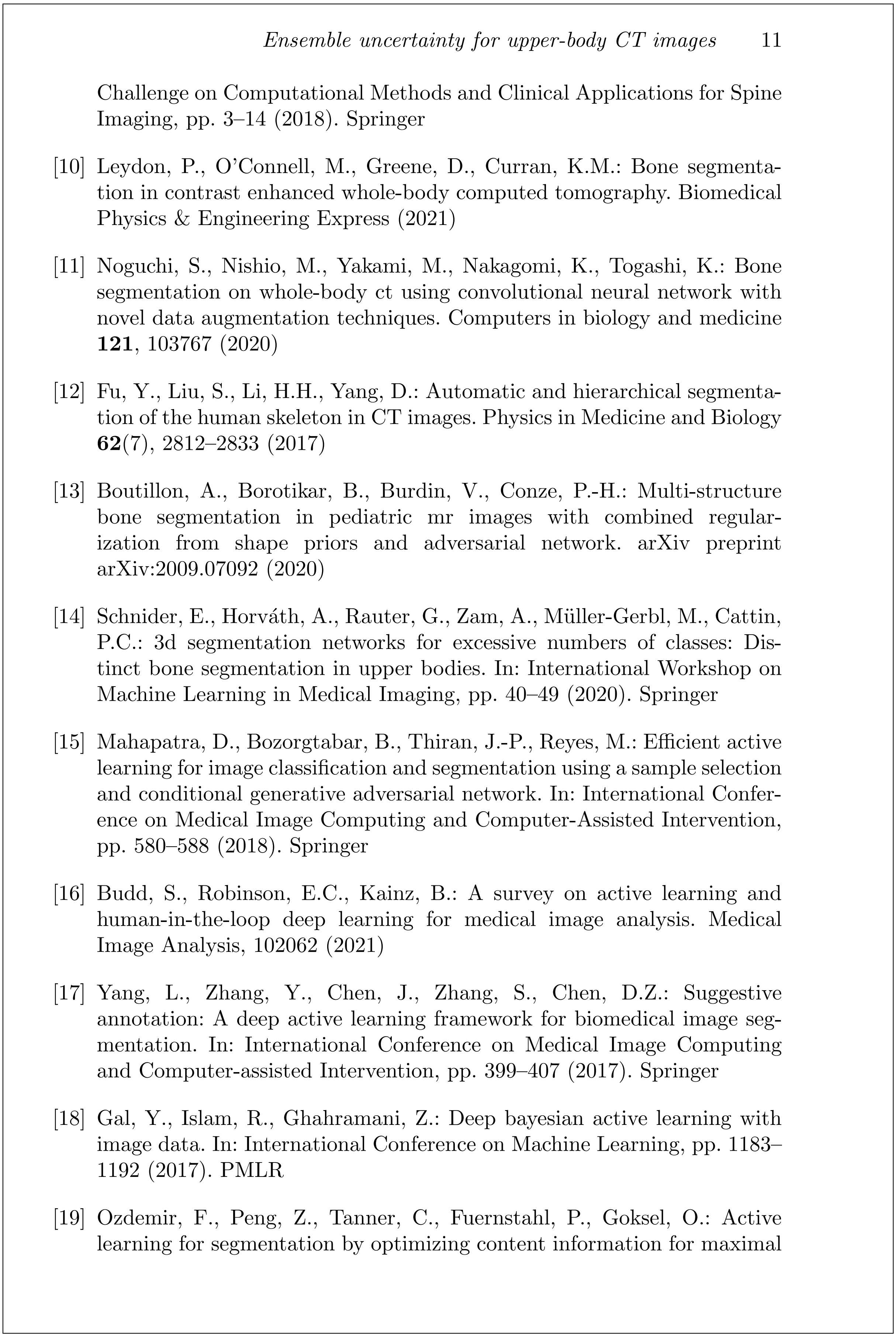
[5] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 234–241 (2015). Springer

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

[7] Gatti, A.A., Maly, M.R.: Automatic knee cartilage and bone segmentation using multi-stage convolutional neural networks: data from the osteoarthritis initiative. Magnetic Resonance Materials in Physics, Biology and Medicine, 1–17 (2021)

[8] Wolleb, J., Sandkühler, R., Cattin, P.C.: Descargan: Disease-specific anomaly detection with weak supervision. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 14–24 (2020). Springer

[9] Horváth, A., Tsagkas, C., Andermatt, S., Pezold, S., Parmar, K., Cattin, P.: Spinal cord gray matter-white matter segmentation on magnetic resonance amira images with md-gru. In: International Workshop and



Challenge on Computational Methods and Clinical Applications for Spine Imaging, pp. 3–14 (2018). Springer

[10] Leydon, P., O'Connell, M., Greene, D., Curran, K.M.: Bone segmentation in contrast enhanced whole-body computed tomography. Biomedical Physics & Engineering Express (2021)

[11] Noguchi, S., Nishio, M., Yakami, M., Nakagomi, K., Togashi, K.: Bone segmentation on whole-body ct using convolutional neural network with novel data augmentation techniques. Computers in biology and medicine **121**, 103767 (2020)

[12] Fu, Y., Liu, S., Li, H.H., Yang, D.: Automatic and hierarchical segmentation of the human skeleton in CT images. Physics in Medicine and Biology **62**(7), 2812–2833 (2017)

[13] Boutillon, A., Borotikar, B., Burdin, V., Conze, P.-H.: Multi-structure bone segmentation in pediatric mr images with combined regularization from shape priors and adversarial network. arXiv preprint arXiv:2009.07092 (2020)

[14] Schnider, E., Horváth, A., Rauter, G., Zam, A., Müller-Gerbl, M., Cattin, P.C.: 3d segmentation networks for excessive numbers of classes: Distinct bone segmentation in upper bodies. In: International Workshop on Machine Learning in Medical Imaging, pp. 40–49 (2020). Springer

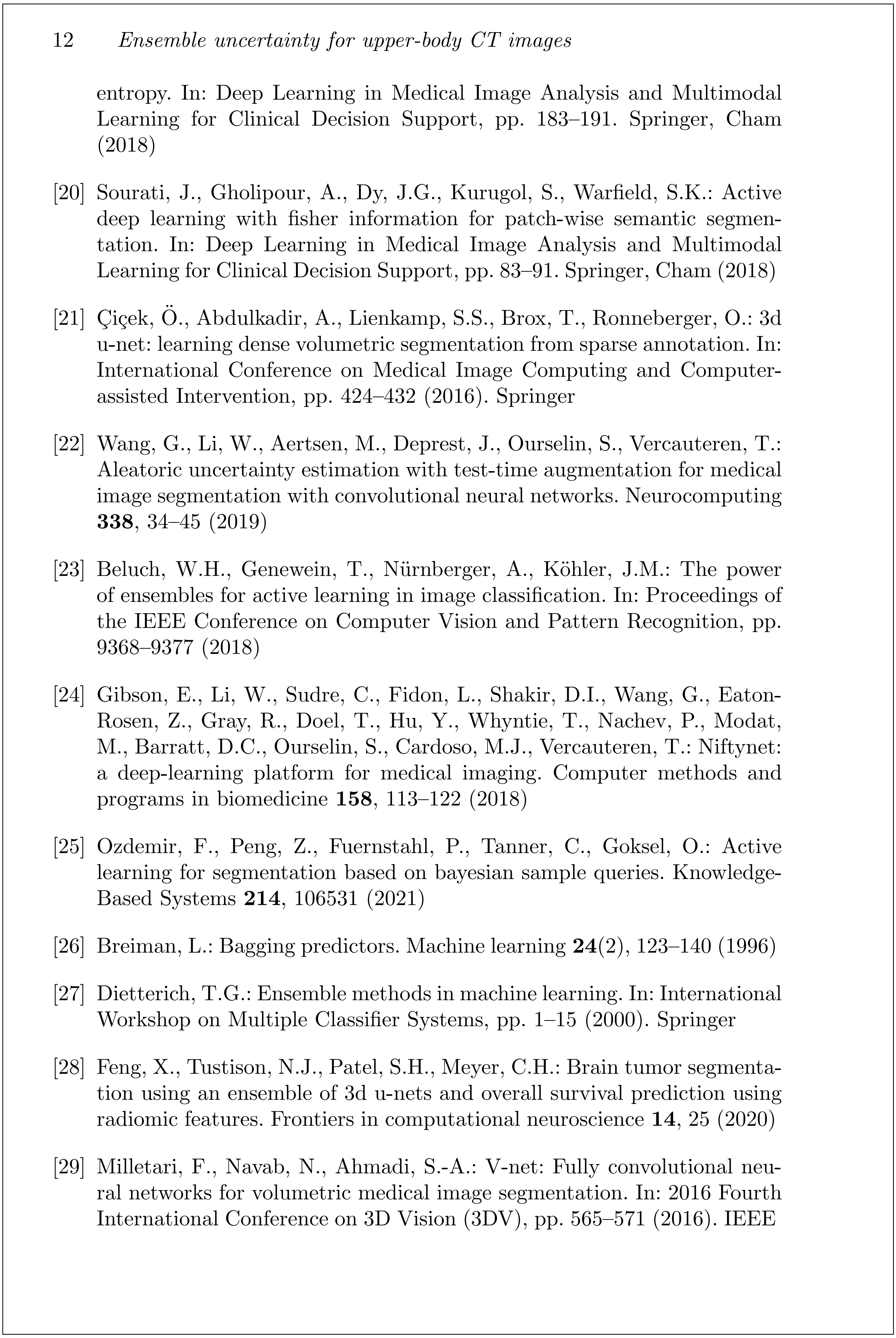
[15] Mahapatra, D., Bozorgtabar, B., Thiran, J.-P., Reyes, M.: Efficient active learning for image classification and segmentation using a sample selection and conditional generative adversarial network. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 580–588 (2018). Springer

[16] Budd, S., Robinson, E.C., Kainz, B.: A survey on active learning and human-in-the-loop deep learning for medical image analysis. Medical Image Analysis, 102062 (2021)

[17] Yang, L., Zhang, Y., Chen, J., Zhang, S., Chen, D.Z.: Suggestive annotation: A deep active learning framework for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 399–407 (2017). Springer

[18] Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: International Conference on Machine Learning, pp. 1183–1192 (2017). PMLR

[19] Ozdemir, F., Peng, Z., Tanner, C., Fuernstahl, P., Goksel, O.: Active learning for segmentation by optimizing content information for maximal



entropy. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, pp. 183–191. Springer, Cham (2018)

[20] Sourati, J., Gholipour, A., Dy, J.G., Kurugol, S., Warfield, S.K.: Active deep learning with fisher information for patch-wise semantic segmentation. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, pp. 83–91. Springer, Cham (2018)

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

[22] Wang, G., Li, W., Aertsen, M., Deprest, J., Ourselin, S., Vercauteren, T.: Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. Neurocomputing **338**, 34–45 (2019)

[23] Beluch, W.H., Genewein, T., Nürnberger, A., Köhler, J.M.: The power of ensembles for active learning in image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9368–9377 (2018)

[24] Gibson, E., Li, W., Sudre, C., Fidon, L., Shakir, D.I., Wang, G., Eaton-Rosen, Z., Gray, R., Doel, T., Hu, Y., Whyntie, T., Nachev, P., Modat, M., Barratt, D.C., Ourselin, S., Cardoso, M.J., Vercauteren, T.: Niftynet: a deep-learning platform for medical imaging. Computer methods and programs in biomedicine **158**, 113–122 (2018)

[25] Ozdemir, F., Peng, Z., Fuernstahl, P., Tanner, C., Goksel, O.: Active learning for segmentation based on bayesian sample queries. Knowledge-Based Systems **214**, 106531 (2021)

[26] Breiman, L.: Bagging predictors. Machine learning **24**(2), 123–140 (1996)

[27] Dietterich, T.G.: Ensemble methods in machine learning. In: International Workshop on Multiple Classifier Systems, pp. 1–15 (2000). Springer

[28] Feng, X., Tustison, N.J., Patel, S.H., Meyer, C.H.: Brain tumor segmentation using an ensemble of 3d u-nets and overall survival prediction using radiomic features. Frontiers in computational neuroscience **14**, 25 (2020)

[29] Milletari, F., Navab, N., Ahmadi, S.-A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571 (2016). IEEE

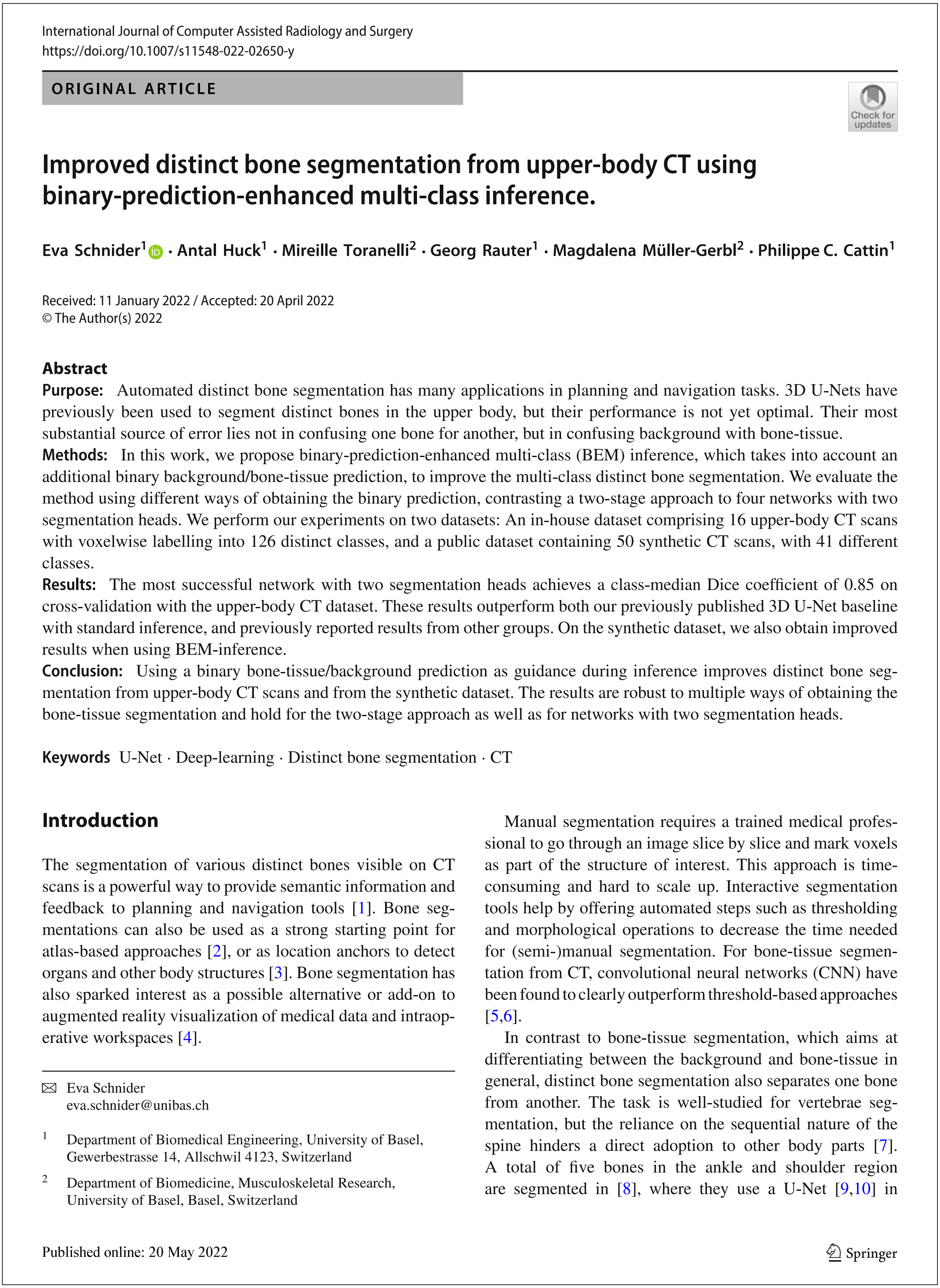
Chapter 7

Improved Distinct Bone Segmentation from Upper-Body CT using Binary-Prediction-Enhanced Multi-Class Inference.

While the previous studies consisted of a proof of concept and the goal to obtain more labelled training data, we now address some of our models' most significant sources of error. Many wrongly classified voxels suffer from a confusion of foreground and background, not so much from a confusion of different foreground bone classes. In this publication we propose a modified inference that combines a binary foreground/background prediction and a multi-class bone prediction. We propose a two-stage approach where two different networks are in charge of making the two predictions and a one-stage approach where the network has two prediction heads that are trained simultaneously. We ablate the network structure and show that the approach works for various architectures. We also propose label correcting post-processing and show that it improves the results on its own and in conjunction with the proposed BEM inference.

Publication. The following manuscript was presented at the conference for *Computer Assisted Radiology and Surgery* (CARS), June 2022, Tokyo, Japan. It was published as part of the conference proceedings [112] in the *International Journal of Computer Assisted Radiology and Surgery* (IJCARS)[1].

[1] https://doi.org/10.1007/s11548-022-02650-y



International Journal of Computer Assisted Radiology and Surgery
https://doi.org/10.1007/s11548-022-02650-y

ORIGINAL ARTICLE

Improved distinct bone segmentation from upper-body CT using binary-prediction-enhanced multi-class inference.

Eva Schnider[1] · Antal Huck[1] · Mireille Toranelli[2] · Georg Rauter[1] · Magdalena Müller-Gerbl[2] · Philippe C. Cattin[1]

Received: 11 January 2022 / Accepted: 20 April 2022
© The Author(s) 2022

Abstract

Purpose: Automated distinct bone segmentation has many applications in planning and navigation tasks. 3D U-Nets have previously been used to segment distinct bones in the upper body, but their performance is not yet optimal. Their most substantial source of error lies not in confusing one bone for another, but in confusing background with bone-tissue.

Methods: In this work, we propose binary-prediction-enhanced multi-class (BEM) inference, which takes into account an additional binary background/bone-tissue prediction, to improve the multi-class distinct bone segmentation. We evaluate the method using different ways of obtaining the binary prediction, contrasting a two-stage approach to four networks with two segmentation heads. We perform our experiments on two datasets: An in-house dataset comprising 16 upper-body CT scans with voxelwise labelling into 126 distinct classes, and a public dataset containing 50 synthetic CT scans, with 41 different classes.

Results: The most successful network with two segmentation heads achieves a class-median Dice coefficient of 0.85 on cross-validation with the upper-body CT dataset. These results outperform both our previously published 3D U-Net baseline with standard inference, and previously reported results from other groups. On the synthetic dataset, we also obtain improved results when using BEM-inference.

Conclusion: Using a binary bone-tissue/background prediction as guidance during inference improves distinct bone segmentation from upper-body CT scans and from the synthetic dataset. The results are robust to multiple ways of obtaining the bone-tissue segmentation and hold for the two-stage approach as well as for networks with two segmentation heads.

Keywords U-Net · Deep-learning · Distinct bone segmentation · CT

Introduction

The segmentation of various distinct bones visible on CT scans is a powerful way to provide semantic information and feedback to planning and navigation tools [1]. Bone segmentations can also be used as a strong starting point for atlas-based approaches [2], or as location anchors to detect organs and other body structures [3]. Bone segmentation has also sparked interest as a possible alternative or add-on to augmented reality visualization of medical data and intraoperative workspaces [4].

Manual segmentation requires a trained medical professional to go through an image slice by slice and mark voxels as part of the structure of interest. This approach is time-consuming and hard to scale up. Interactive segmentation tools help by offering automated steps such as thresholding and morphological operations to decrease the time needed for (semi-)manual segmentation. For bone-tissue segmentation from CT, convolutional neural networks (CNN) have been found to clearly outperform threshold-based approaches [5,6].

In contrast to bone-tissue segmentation, which aims at differentiating between the background and bone-tissue in general, distinct bone segmentation also separates one bone from another. The task is well-studied for vertebrae segmentation, but the reliance on the sequential nature of the spine hinders a direct adoption to other body parts [7]. A total of five bones in the ankle and shoulder region are segmented in [8], where they use a U-Net [9,10] in

✉ Eva Schnider
eva.schnider@unibas.ch

[1] Department of Biomedical Engineering, University of Basel, Gewerbestrasse 14, Allschwil 4123, Switzerland

[2] Department of Biomedicine, Musculoskeletal Research, University of Basel, Basel, Switzerland

Published online: 20 May 2022

combination with shape priors and adversarial regularization. They also compare the performance of separate U-Nets trained on one bone class each versus a multi-class U-Net which outperformed the combined single-class networks.

Segmentation into a larger number of distinct bones has not yet been investigated in many cases. A hierarchical atlas-based approach leads to good segmentation results of 62 distinct bones from upper-body CTs at the expense of a long inference time [2]. In [11], 49 distinct bone classes have been segmented on upper-body CTs. They used a two-stage approach where a landmark detection network was followed by a voxelwise segmentation by a dilation-based CNN and the deletion of all but the largest connected component per class. Neither of these two approaches offers an end-to-end method or includes the bones of the hand in the segmentation. A segmentation that also includes these bones, totalling to 126 bone classes, has been investigated on a smaller dataset in one of our previous works [12], where we found a 3D U-Net to be better suited to the task than the 2D U-Nets commonly used in a slicewise way for bone-tissue segmentation.

The purpose of of this current work is to reduce the most prevalent segmentation errors of the 3D U-Net when performing distinct bone segmentation. To do so, we propose to leverage an additional binary segmentation during the inference process. A related approach has been examined by [13] who combine the outputs of a semantic segmentation head and an instance segmentation head into a panoptic segmentation for 2D traffic images. Apart from the dimensionality and the image modality, our work also differs as we stay within a semantic segmentation problem statement.

We propose and investigate BEM, an inference method that enhances a multi-class distinct bone segmentation using a binary bone-tissue/background segmentation. We compare the segmentation accuracy, run-time, and complexity of different network architectures that achieve both segmentations within a single trained model, and contrast the results to a two-stage approach.

Materials and methods

Upper-body CT dataset

Our in-house dataset consists of 17 upper-body CT scans, and corresponding voxelwise segmentations created by specialists, with an isotropic resolution of 2 mm , as used in [14]. The dataset comprises postmortem scans of 9 male and 7 female body donours aged 44–103 years. Before resampling, the scans were of varying resolution with slightly less than 1 mm resolution in-plane and up to 1.5 mm out-plane. Due to inconsistent arm positioning, we excluded one scan from the set in this work. The segmentation contains 126 different classes, including background (Fig. 1).

Synthetic 3D dataset

We created a synthetic dataset in order to highlight the effect of the proposed BEM-inference on anatomical segmentation tasks and to provide results on a publicly available dataset (published at https://gitlab.com/cian.unibas.ch/cars2022-bem-inference). The dataset was constructed by generating a randomly varying three-dimensional stick-figure-like ground truth segmentation consisting of 41 distinct bones (see Fig. 2). Inspired by human anatomy, we chose similar geometric shapes for similar bones such as vertebrae, to force the networks to rely not only on shapes but also the relative positioning of structures. To construct the soft-tissue

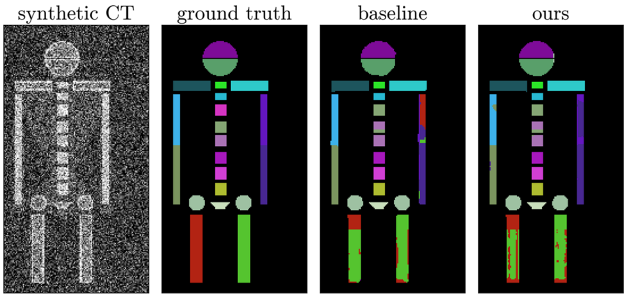


Fig. 2 Results on the synthetic dataset using the baseline 3D U-Net, and Dual D with our proposed BEM-inference. Both false positives (around the elbows), and false negatives (head) are reduced using our approach

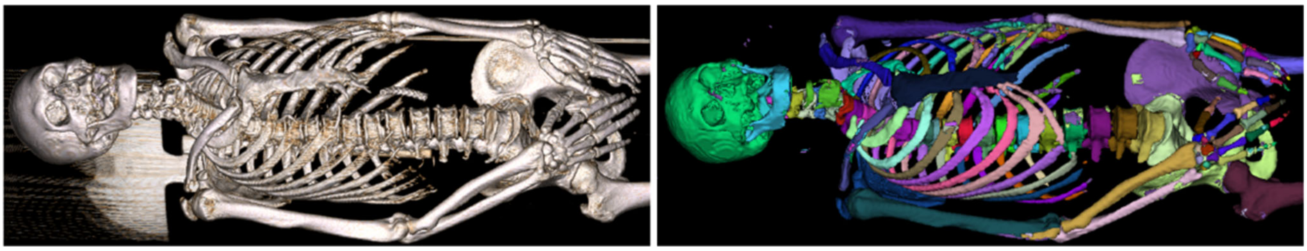


Fig. 1 Volume rendering of one of our upper-body CT scans (left), and the result of our automated segmentation using BEM-inference and label-correction (right)

area, we created convex hulls for the torso, limbs, and head. Finally, we filled areas of background, soft-tissue, cortical bone and cancellous bone with typical HU-values and added uniform random noise. Emphasis is not put on the anatomical accuracy of the dataset, but on the ability to mimic the difficulty of our primary task, which is to study the simultaneous detection and distinction of many three-dimensional structures with groupwise similar shapes.. The final synthetic CT scans measure 128 × 128 × 256 voxels.

Base architecture

We use an architecture based on the 3D U-Net [10], which is composed of a decoder and encoder with skip connections. Following [15], we add instance normalization, use leaky rectified linear units (leReLU) and exchange the upconvolutions in favor of linear upsampling. The high computational demand of a 3D network with a large number of classes, restricts the possible batch size to one. We implemented the network in Tensorflow-Keras 2.5.

Dual segmentation head architecture

To obtain the multi-class and the binary background/bone-tissue segmentation simultaneously, we explore four architectures with two segmentation heads. A comparison of their architectures is given in Table 1 and Fig. 3.

- **Dual A** All layers except the classification heads are shared.
- **Dual B** Both tasks still share the whole encoder and decoder but have their own convolutional layers at full resolution.
- **Dual C** Both tasks share the full encoder and decoder. The binary segmentation head is appended after the decoder, the distinct bone segmentation head follows after one more convolutional block at full resolution.
- **Dual D** Both tasks share the encoder and feature encoding, but have their own decoders.

Table 1 Network architectures comparison for the upper-body CT dataset

Model	Trainable parameters (#)	Training time [1](s)	Inference time [2] (s)
Baseline 3D U-Net	$1.46 \cdot 10^7$	0.84	219
Dual A	$1.46 \cdot 10^7$	1.08	212
Dual B	$1.46 \cdot 10^7$	1.08	271
Dual C	$1.46 \cdot 10^7$	1.15	243
Dual D	$1.98 \cdot 10^7$	1.20	321

[1] Average time per training iteration on a 64^3 voxel patch.
[2] Inference time for an average scan (∼ 256 × 256 × 512 voxels) , including data I/O time

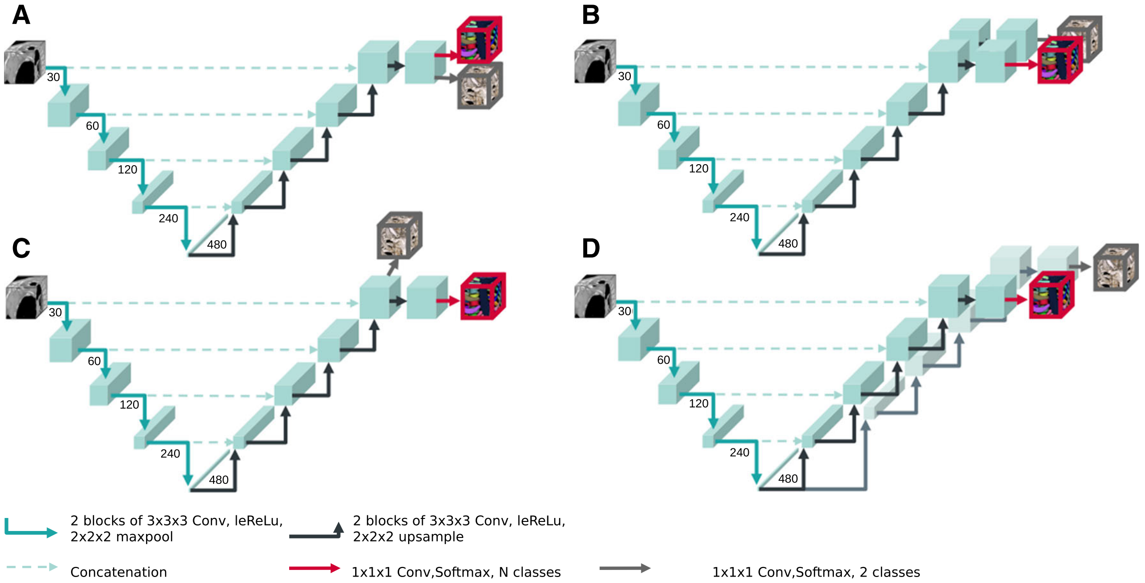

A
B
C
D
30
60
120
240
480
2 blocks of 3x3x3 Conv, leReLu, 2x2x2 maxpool
2 blocks of 3x3x3 Conv, leReLu, 2x2x2 upsample
Concatenation
1x1x1 Conv,Softmax, N classes
1x1x1 Conv,Softmax, 2 classes

Fig. 3 Schematic of the four network architectures with dual segmentation heads. They are all based on a 3D U-Net architectures with variances of how the binary segmentation head is appended. See also "Dual segmentation head architecture" Section

Two-stage approach

As an alternative to the architectures with dual segmentation heads, we study the results using a binary prediction, which is obtained separately from the full multi-class network. To do so, we train an additional instance of our baseline 3D U-Net on the background/bone-tissue problem alone and use the resulting binary segmentation during the BEM-inference step. As an upper bound, we also compute results using the ground truth of the binary segmentation.

Training and standard inference

For both datasets, we optimize our networks using the Adam optimizer with a learning rate of 0.001 for 75000 iterations, after which all of our models had converged. Total training time is roughly one day per cross-validation fold on one GeForce GTX Titan X (12 GB). We use five cross-validation splits for the upper-body dataset, where we use 11 scans for training, 2 for validation of the convergence, and 3 for testing. For the synthetic dataset, we were able to create a larger number of validation and test images to get more representative test results and thus evaluate one fold only. We use 17 volumes for training, 7 for validation, and 26 for testing.

As loss function we use an unweighted combination of the cross-entropy loss $\mathcal{L}_{\text{X-Ent}}$ and the Dice loss $\mathcal{L}_{\text{DSC}}$ [16]. In the dual segmentation head networks, we add the losses for the binary background/bone-tissue task:

$$\mathcal{L}_{\text{total}} := \mathcal{L}^{C}_{\text{X-Ent}} + \sum_{c \in C} \mathcal{L}^{c}_{\text{DSC}} + \mathcal{L}^{\{\text{bg},\text{bt}\}}_{\text{X-Ent}} + \sum_{c \in \{\text{bg},\text{bt}\}} \mathcal{L}^{c}_{\text{DSC}}$$

We train our network patchwise since the use of whole CT volumes for training is not computationally feasible in 3D. The patch size not only influences the computational requirements, but also the network accuracy [17]. We found a patch size of 64^3 voxels to be a good compromise. The patchwise sampling also serves as a random-cropping data-augmentation step. Other common data augmentation techniques such as rotations, scaling, or mix-up are not used in this work. Data augmentation has been studied in-depth for whole-body bone-tissue segmentation, where it only leads to very small improvements [5].

Prior to inference, we pad our scans by 20 voxels to mitigate the proximity of the hands to the image border in some of the scans. After padding, our predictions are assembled using a sliding window approach with a 20 voxel overlap to increase the influence of the centre of the patches on the final predictions, which has been shown to lead to good results [15]. The voxelwise multi-class prediction is conducted by a softmax activation.

BEM-inference

We refine the inference step using a binary background/bone-tissue segmentation $y_{\text{bg/bt}}$. This additional prediction can stem from a second head of the multi-class network, from an additional network, or from a completely different segmentation method.

In standard inference, all classes, including the background class, are predicted in one step. Instead, we use the binary prediction $y_{\text{bg/bt}}$ as a guide and ignore the background class 0 in the distinct bone prediction. We split our N classes into one background and $N-1$ foreground classes. The final prediction is then set to be either background, if $y_{\text{bg/bt}} = 0$ or to the most likely foreground class.

In contrast to simple masking of the finished multi-class prediction in post-processing, which could remove false negative foreground voxels, this method addresses both false negatives and false positives. An illustration of a simplified case in 2D with two foreground classes can be found in Fig. 4.

Connected component-based label correction

After completion of the inference process, we automatically refine the segmentation by reassigning connected components. We build upon the post-processing approach of keeping only the biggest connected component per label [11]. However, instead of assigning all smaller components to the background, we assign them to their neighboring biggest component. To do so, we define sets of bones that are easily confused by a model. Within such a set L, we identify all connected components per class and choose its largest connected component as the class anchor. Adjacent smaller components of other classes are then reassigned the anchor label. The sets L are chosen based on anatomical knowledge and on the most frequent confusions among bone classes observed on the validation set. To save-guard against very fragmented segmentations, an upper threshold u of connected components ensures a runtime of $\mathcal{O}(|L|^2 u)$. Different sets can be processed in parallel to speed up the computation. We chose $u = 100$ and worked with 16 sets L, of size $4 \le |L| \le 12$. The detailed groups are shared along with the code at https://gitlab.com/cian.unibas.ch/cars2022-bem-inference.

Evaluation metrics

As our main metric, we use the Sørensen-Dice similarity coefficient DSC_c for each segmentation class c. To assess the overall performance of our models, we give the median, and the 16- and 84-percentile ($\sim 1\sigma$) of all classes where at least one true-positive voxel has been predicted as median$^{+\sigma}_{-\sigma}$. We account for the remaining classes, those with $\text{DSC}_\text{c} = 0$, by providing the fraction of classes where $\text{DSC}_\text{c} > 0$ in brackets. We account for the completely missing classes by providing the fraction of detected classes in brackets.

Results and discussion

Our results show how a BEM-inference combined with connected-component correcting post-processing can improve automated distinct bone segmentation from upper-body CTs. Our evaluation involves two different datasets, four flavors of U-Nets with dual segmentation heads, and a two-stage approach.

Test We evaluated the errors most commonly experienced while conducting a baseline U-Net segmentation on our upper-body CT dataset. The confusion matrix (Fig. 5, left, first column) illustrates our finding, that many errors originate from predicting bones as background, as opposed to confusing one bone for another. This type of error is reduced when using our proposed methods (Fig. 5, right, first column).

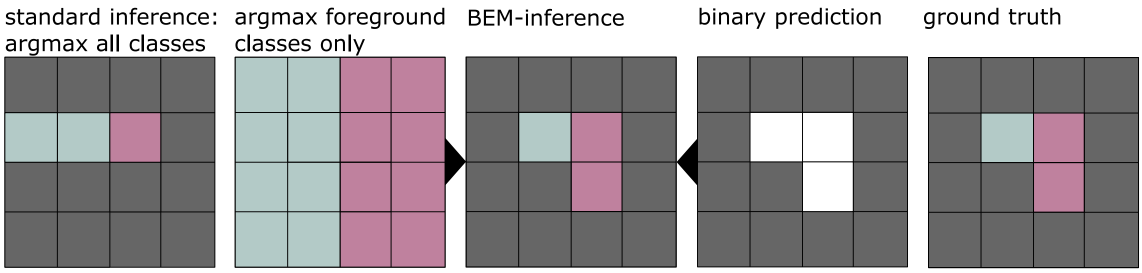

standard inference: argmax all classes
argmax foreground classes only
BEM-inference
binary prediction
ground truth

Fig. 4 Schematic of the BEM-inference process. The background class is denoted in gray, the two distinct foreground classes in blue and pink, respectively

Fig. 5 Label confusion matrices (row-normalized) for the baseline 3D U-Net and Dual D, including BEM-inference and post-processing. With our approach, less labels are erroneously classified as background (first column)

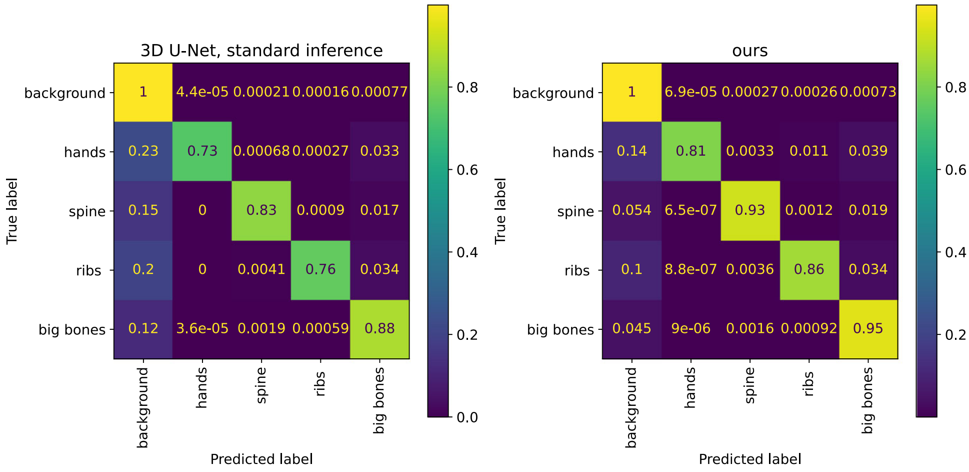


Table 2 Upper-body CT dataset: Results in DSC, comparing the segmentation performance when using baseline inference, against our BEM-inference, with and without label correction

	Baseline	+ Label correction	+ BEM-inference	+ Both
Baseline 3D U-Net	$0.78^{+0.12}_{-0.29}$, (0.95)	$0.81^{+0.09}_{-0.25}$, (0.94)		
Two-stage: pred. bin.	”	”	$0.79^{+0.11}_{-0.30}$, (0.96)	$0.82^{+0.09}_{-0.26}$, (0.94)
Two-stage: gt bin.	”	”	$0.89^{+0.08}_{-0.29}$, (0.96)	$0.93^{+0.05}_{-0.22}$, (0.95)
Dual A	$0.78^{+0.11}_{-0.29}$, (0.96)	$0.81^{+0.09}_{-0.27}$, (0.95)	$0.79^{+0.11}_{-0.30}$, (0.97)	$0.82^{+0.10}_{-0.27}$, (0.95)
Dual B	$0.77^{+0.12}_{-0.28}$, (0.95)	$0.81^{+0.09}_{-0.29}$, (0.94)	$0.79^{+0.11}_{-0.30}$, (0.96)	$0.82^{+0.09}_{-0.28}$, (0.95)
Dual C	$0.79^{+0.10}_{-0.31}$, (0.96)	$0.82^{+0.09}_{-0.28}$, (0.95)	$0.79^{+0.11}_{-0.31}$, (0.96)	$0.82^{+0.09}_{-0.29}$, (0.95)
Dual D	$0.80^{+0.10}_{-0.29}$, (0.95)	$0.84^{+0.08}_{-0.24}$, (0.94)	$0.82^{+0.11}_{-0.29}$, (0.96)	$0.85^{+0.08}_{-0.24}$, (0.94)

The comparison is given for the two-stage models and the different flavors of dual-segmentation heads models. For a description of the metrics, see “Evaluation metrics” Section

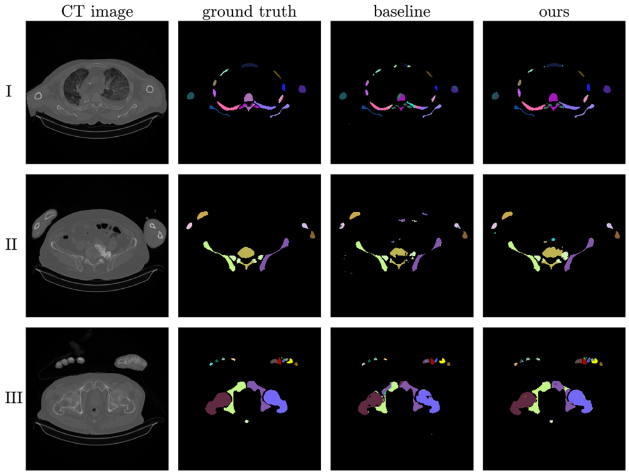


Fig. 6 Segmentation results and typical errors obtained with the baseline U-Net model and our Dual D model with BEM-inference and post-processing. Using the baseline model, ribs are often not segmented as one, but are assigned multiple labels (I). The post-processing remedies this issue visibly. Other frequent errors occur around the border of vertebrae, especially in the presence of calcifications (II). Within big bones such as hips and femurs, we observe holes and islands where the left/right part of the label has been mixed up (III)

We conducted an ablation study on the upper-body CT dataset, where we examined the influence of how the binary prediction was created (two-stage versus networks with dual segmentation heads), the network architecture, and the label correction post-processing. The results are listed in Table 2. Common errors are illustrated in Fig. 6. The proposed method using a Dual D model, BEM-inference and the post-processing label correction detected correct voxels in 94% of all bones and achieved a median DSC of 0.85, which is an improvement over our baseline with a median of 0.78. Both the BEM-inference and post-processing contribute individually to the improved DSC scores, but the strongest results are achieved in combination.

We observe a small increase of the fraction of bone classes with DSC > 0 when using the enhanced inference, and a slight decrease when using the post-processing. The majority of classes with a DSC of 0 are small bones located in the hands.

In Table 4, we compare our results to the hierarchical atlas segmentation by Fu et al. [2] and the convolutional neural networks by Lindgren Belal et al. [11]. Our results compete well, although the use of different datasets hampers a direct comparison.

Among the models with two segmentation heads, the most complex version Dual D with two separate decoders led to the best results. Merely training two decoders simultaneously on two different loss functions led to first improvements over our baseline, which improved even further when using BEM-inference and label-correction.

The results of the two-stage approach depend on the performance of both the multi-class and binary segmentation model. We used a binary segmentation predicted by the baseline 3D U-Net trained on the background/bone-tissue segmentation task. This network achieved a mean DSC of 0.94 for the binary prediction, which is in the range of results reported in [5] and [6]. For comparison, we used the binary ground truth data during the BEM-inference step to get an

Table 3 Synthetic dataset: Results in DSC, comparing the segmentation performance when using baseline inference, against our BEM-inference, with and without label correction

Model	Baseline	+ BEM-inference
Two-stage: gt binary seg.	$0.973^{+0.030}_{-0.240}$, (1.00)	$0.991^{+0.010}_{-0.250}$, (1.00)
Dual A: parallel losses	$0.970^{+0.030}_{-0.230}$, (1.00)	$0.970^{+0.030}_{-0.230}$, (1.00)
Dual B: parallel final layers	$0.971^{+0.030}_{-0.230}$, (0.99)	$0.978^{+0.020}_{-0.230}$, (0.99)
Dual C: sequential heads	$0.963^{+0.040}_{-0.260}$, (0.99)	$0.966^{+0.030}_{-0.250}$, (0.99)
Dual D: separate decoders	$0.975^{+0.020}_{-0.230}$, (1.00)	$0.982^{+0.020}_{-0.230}$, (1.00)

The comparison is given for the two-stage models and the different flavors of dual-segmentation heads models. For a description of the metrics, see “Evaluation metrics” Section

Table 4 Comparison to other published work on distinct bone segmentation

	Ours (median)	[11] (median)	[2] (mean)
L3	0.85	0.85	0.91
Sacrum	0.90	0.88	
Clavicula	0.92		0.57
Hamate	0.86		
Inference time per scan (min)	∼ 5		∼ 20
Scans in dataset (#)	11	100	19
Classes (#)	126	49	62

Results in DSC

upper bound of how much improvement was possible. We observed a steep improvement of the results, suggesting that the investment into a good binary segmentation clearly pays off. Since the manual labelling of the ground truth data is less time-consuming and cumbersome for the binary segmentation as opposed to a full multi-class segmentation, the additional binary labelling of new training data might yield a good return on investment.

In comparison, the two-stage approach tends to be more troublesome than a dual head architecture since it involves the training and tuning of two networks and a sequential inference first using the binary network, then the multi-class network. The use of a network with two segmentation heads simplifies this task to training one network only and performing an end-to-end inference. If additional scans with binary ground truth labelling are available, they can be used to fine-tune the binary segmentation head.

There is currently no public upper-body CT dataset with complete distinct bone labelling available and our in-house dataset cannot be shared as of yet. Therefore, we provided additional results on our public synthetic dataset. The results on the synthetic dataset mirror the findings in the upper-body dataset. BEM-inference improves the segmentation both for the two-stage approach and the architectures with dual segmentation heads (see Table 3 and Fig. 2).

Conclusion

We proposed BEM-inference to improve the automated segmentation of distinct bones from upper-body CT scans. A substantial part of the segmentation errors made by 3D U-Nets does not originate from the mixing-up of different bone classes but from the mistaking of background for the foreground , and vice versa. Therefore, we proposed an inference method that uses the information gained in a binary background/bone-tissue segmentation to improve upon the multi-class inference. We compared two approaches to obtain the necessary binary segmentation: (1) Networks with dual segmentation heads that are trained on both tasks simultaneously, (2) and a two-stage approach where separate networks are trained for the multi-class and the binary segmentation task. Using our proposed inference lead to improvements on all architectures and on both datasets, with and without our label-correction post-processing . The class-median DSC of the dual decoder network with both post-processing and BEM-inference is 0.85 on the upper-body CT dataset, outperforming the baseline 3D U-Net and previously reported results by other groups.

Our proposed BEM-inference is most suitable for tasks where the binary task is simpler to solve or binary labelled data is easier to obtain than the full multi-class labelled data. Since an existing multi-class ground truth segmentation can easily be converted to a binary ground truth segmentation, any multi-class model can be retrofitted to use two-stage BEM-inference. if a source of binary segmentations is available or trainable This makes BEM-inference a versatile addition to anatomical multi-class segmentation workflows.

Acknowledgements This work was financially supported by the Werner Siemens Foundation through the MIRACLE project. We thank Azhar Zam for valuable discussions that helped shape this work.

Funding Open access funding provided by University of Basel. This work was financially supported by the Werner Siemens Foundation through the MIRACLE project.

Code availability Code for post-processing and to generate the synthetic dataset is shared at: https://gitlab.com/cian.unibas.ch/cars2022-bem-inference.

Declarations

Conflict of interest None of the authors have competing interests to declare that are relevant to the content of this article.

Consent to participate Informed consent was obtained from all individual body donours included in the study.

Consent for publication Body donours signed informed consent regarding publications using their data.

Ethical approval This research study was conducted retrospectively from CT data routinely obtained from body donours. No ethical approval is required.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Qiu B, Guo J, Kraeima J, Glas HH, Borra RJ, Witjes MJ, van Ooijen PM (2019) Automatic segmentation of the mandible from computed tomography scans for 3d virtual surgical planning using the convolutional neural network. Phys Med Biol 64(17):175020
2. Fu Y, Liu S, Li HH, Yang D (2017) Automatic and hierarchical segmentation of the human skeleton in CT images. Phys Med Biol 62(7):2812–2833
3. Kamiya N, Kume M, Zheng G, Zhou X, Kato H, Chen H, Muramatsu C, Hara T, Miyoshi T, Matsuo M, Fujita H (2018) Automated recognition of erector spinae muscles and their skeletal attachment region via deep learning in torso ct images. International

workshop on computational methods and clinical applications in musculoskeletal imaging. Springer, Cham, pp 1–10

4. Żelechowski M, Karnam M, Faludi B, Gerig N, Rauter G, Cattin PC (2021) Patient positioning by visualising surgical robot rotational workspace in augmented reality. Comput Methods Biomech Biomed Eng Imaging Vis, 1–7
5. Noguchi S, Nishio M, Yakami M, Nakagomi K, Togashi K (2020) Bone segmentation on whole-body ct using convolutional neural network with novel data augmentation techniques. Comput Biol Medicine 121:103767
6. Klein A, Warszawski J, Hillengaß J, Maier-Hein KH (2019) Automatic bone segmentation in whole-body ct images. Int J Comput Assist Radiol Surg 14(1):21–29
7. Cheng P, Yang Y, Yu H, He Y (2021) Automatic vertebrae localization and segmentation in ct with a two-stage dense-u-net. Sci Rep 11(1):1–13
8. Boutillon A, Borotikar B, Burdin V, Conze P-H (2020) Multi-structure bone segmentation in pediatric mr images with combined regularization from shape priors and adversarial network. arXiv preprint arXiv:2009.07092
9. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention, pp 234–241. Springer
10. Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O (2016) 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention, pp 424–432 . Springer
11. Lindgren Belal S, Sadik M, Kaboteh R, Enqvist O, Ulén J, Poulsen MH, Simonsen J, Høilund-Carlsen PF, Edenbrandt L, Trägårdh E (2019) Deep learning for segmentation of 49 selected bones in CT scans: first step in automated PET/CT-based 3D quantification of skeletal metastases. Eur J Radiol 113:89–95
12. Schnider E, Horváth A, Rauter G, Zam A, Müller-Gerbl M, Cattin PC (2020) 3d segmentation networks for excessive numbers of classes: distinct bone segmentation in upper bodies. In: International workshop on machine learning in medical imaging, pp 40–49 . Springer
13. Cheng B, Collins MD, Zhu Y, Liu T, Huang TS, Adam H, Chen L-C (2020) Panoptic-deeplab: a simple, strong, and fast baseline for bottom-up panoptic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12475–12485
14. Schnider E, Huck A, Rauter G, Zam A, Müller-Gerbl M, Cattin PC Ensemble uncertainty as a criterion for dataset expansion in distinct bone segmentation from upper-body ct images. In: Under submission
15. Isensee F, Jaeger PF, Kohl SA, Petersen J, Maier-Hein KH (2021) nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nat Methods 18(2):203–211
16. Milletari F, Navab N, Ahmadi S-A (2016) V-net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV), pp 565–571. IEEE
17. Suzuki Y, Hori M, Kido S, Otake Y, Ono M, Tomiyama N, Sato Y (2021) Comparative study of vessel detection methods for contrast enhanced computed tomography: effects of convolutional neural network architecture and patch size. Adv Biomed Eng 10:138–149

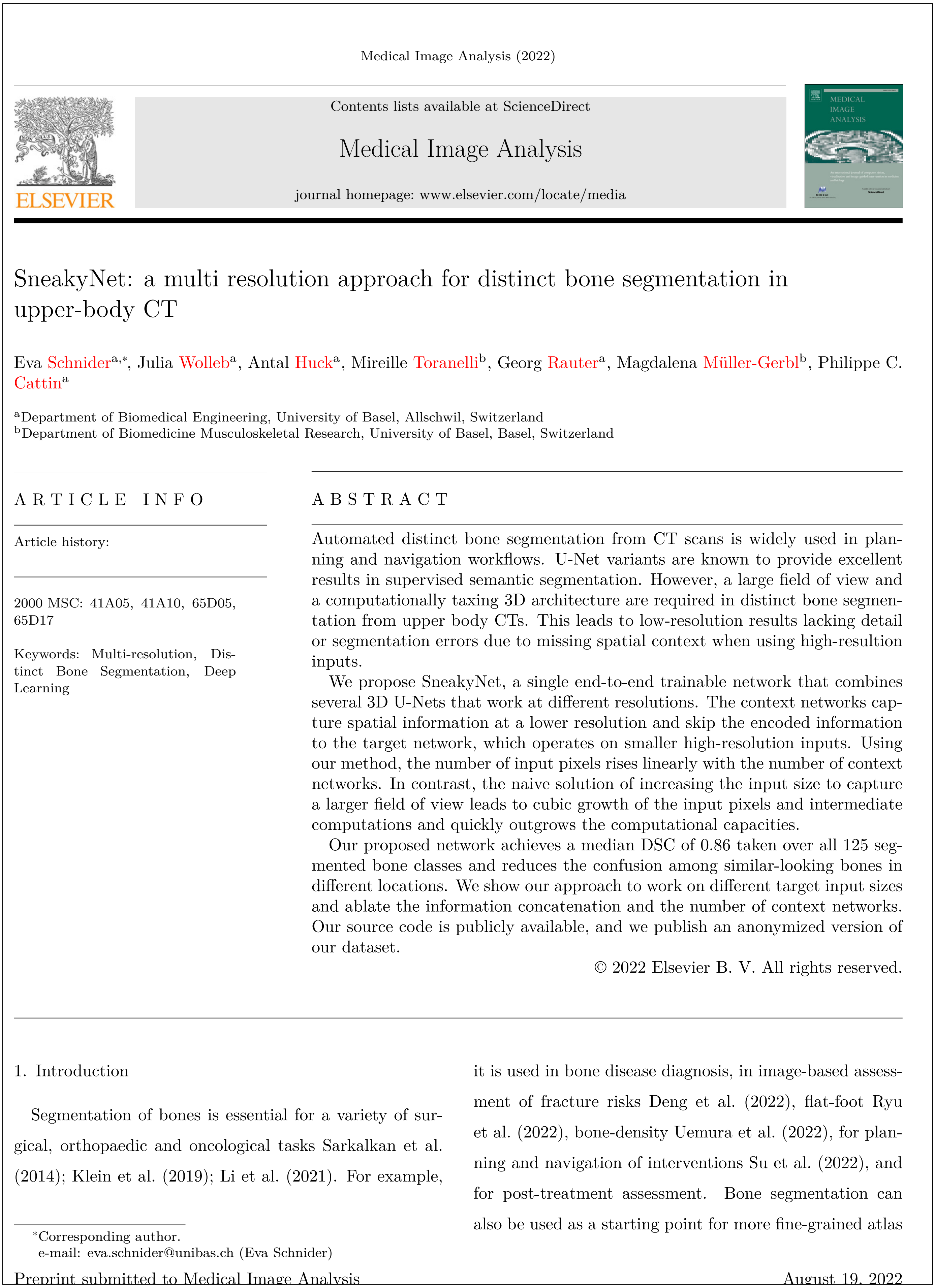
Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Chapter 8

SneakyNet: a Multi-Resolution Approach for Distinct Bone Segmentation in Upper-Body CT.

After dealing with the previous paper's background/foreground confusion issues, we now work on another frequently occurring problem. Specifically, when using smaller input sizes, we realized that many bones of similar shapes were confused with one another, despite having very different locations within the scan. Naively, the easiest way would be just to increase the input size. Since our problem is three dimensional this is only feasible to some extent. Instead, we propose a multi-resolution approach, where we use inputs covering successively bigger fields of view while keeping the number of input voxels constant by using a smaller resolution. We show that this helps obtain better results, particularly when using small input sizes.

Publication. The manuscript has been submitted to the journal *Medical Image Analysis* (MedIA) on the 24th of August 2022 and is currently under review.



Medical Image Analysis (2022)

Contents lists available at ScienceDirect

Medical Image Analysis

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

SneakyNet: a multi resolution approach for distinct bone segmentation in upper-body CT

Eva Schnider[a,*], Julia Wolleb[a], Antal Huck[a], Mireille Toranelli[b], Georg Rauter[a], Magdalena Müller-Gerbl[b], Philippe C. Cattin[a]

[a]Department of Biomedical Engineering, University of Basel, Allschwil, Switzerland
[b]Department of Biomedicine Musculoskeletal Research, University of Basel, Basel, Switzerland

ARTICLE INFO

Article history:

2000 MSC: 41A05, 41A10, 65D05, 65D17

Keywords: Multi-resolution, Distinct Bone Segmentation, Deep Learning

ABSTRACT

Automated distinct bone segmentation from CT scans is widely used in planning and navigation workflows. U-Net variants are known to provide excellent results in supervised semantic segmentation. However, a large field of view and a computationally taxing 3D architecture are required in distinct bone segmentation from upper body CTs. This leads to low-resolution results lacking detail or segmentation errors due to missing spatial context when using high-resultion inputs.

We propose SneakyNet, a single end-to-end trainable network that combines several 3D U-Nets that work at different resolutions. The context networks capture spatial information at a lower resolution and skip the encoded information to the target network, which operates on smaller high-resolution inputs. Using our method, the number of input pixels rises linearly with the number of context networks. In contrast, the naive solution of increasing the input size to capture a larger field of view leads to cubic growth of the input pixels and intermediate computations and quickly outgrows the computational capacities.

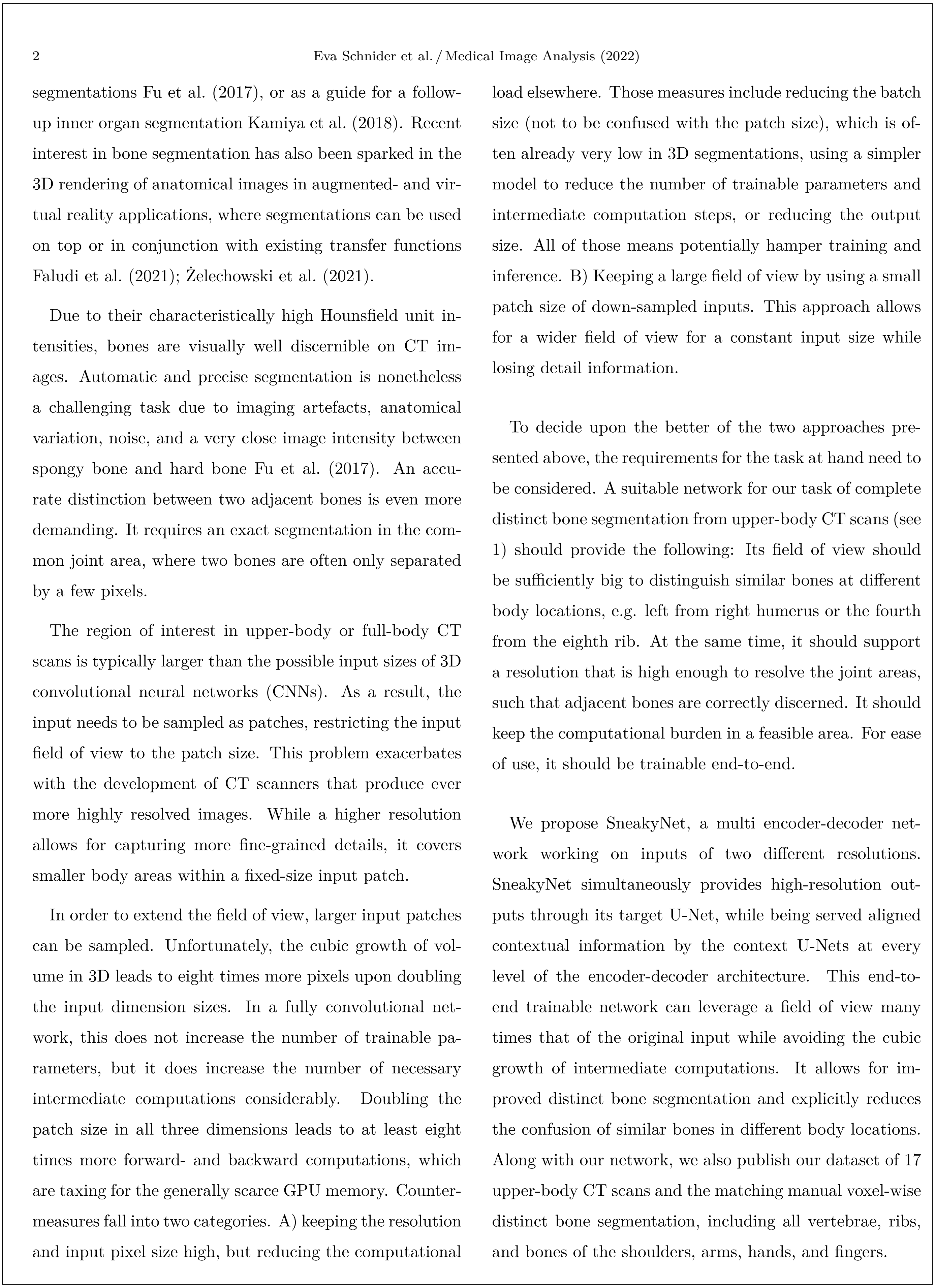
Our proposed network achieves a median DSC of 0.86 taken over all 125 segmented bone classes and reduces the confusion among similar-looking bones in different locations. We show our approach to work on different target input sizes and ablate the information concatenation and the number of context networks. Our source code is publicly available, and we publish an anonymized version of our dataset.

© 2022 Elsevier B. V. All rights reserved.

1. Introduction

Segmentation of bones is essential for a variety of surgical, orthopaedic and oncological tasks Sarkalkan et al. (2014); Klein et al. (2019); Li et al. (2021). For example, it is used in bone disease diagnosis, in image-based assessment of fracture risks Deng et al. (2022), flat-foot Ryu et al. (2022), bone-density Uemura et al. (2022), for planning and navigation of interventions Su et al. (2022), and for post-treatment assessment. Bone segmentation can also be used as a starting point for more fine-grained atlas

[*]Corresponding author.
e-mail: eva.schnider@unibas.ch (Eva Schnider)



segmentations Fu et al. (2017), or as a guide for a follow-up inner organ segmentation Kamiya et al. (2018). Recent interest in bone segmentation has also been sparked in the 3D rendering of anatomical images in augmented- and virtual reality applications, where segmentations can be used on top or in conjunction with existing transfer functions Faludi et al. (2021); Żelechowski et al. (2021).

Due to their characteristically high Hounsfield unit intensities, bones are visually well discernible on CT images. Automatic and precise segmentation is nonetheless a challenging task due to imaging artefacts, anatomical variation, noise, and a very close image intensity between spongy bone and hard bone Fu et al. (2017). An accurate distinction between two adjacent bones is even more demanding. It requires an exact segmentation in the common joint area, where two bones are often only separated by a few pixels.

The region of interest in upper-body or full-body CT scans is typically larger than the possible input sizes of 3D convolutional neural networks (CNNs). As a result, the input needs to be sampled as patches, restricting the input field of view to the patch size. This problem exacerbates with the development of CT scanners that produce ever more highly resolved images. While a higher resolution allows for capturing more fine-grained details, it covers smaller body areas within a fixed-size input patch.

In order to extend the field of view, larger input patches can be sampled. Unfortunately, the cubic growth of volume in 3D leads to eight times more pixels upon doubling the input dimension sizes. In a fully convolutional network, this does not increase the number of trainable parameters, but it does increase the number of necessary intermediate computations considerably. Doubling the patch size in all three dimensions leads to at least eight times more forward- and backward computations, which are taxing for the generally scarce GPU memory. Countermeasures fall into two categories. A) keeping the resolution and input pixel size high, but reducing the computational load elsewhere. Those measures include reducing the batch size (not to be confused with the patch size), which is often already very low in 3D segmentations, using a simpler model to reduce the number of trainable parameters and intermediate computation steps, or reducing the output size. All of those means potentially hamper training and inference. B) Keeping a large field of view by using a small patch size of down-sampled inputs. This approach allows for a wider field of view for a constant input size while losing detail information.

To decide upon the better of the two approaches presented above, the requirements for the task at hand need to be considered. A suitable network for our task of complete distinct bone segmentation from upper-body CT scans (see 1) should provide the following: Its field of view should be sufficiently big to distinguish similar bones at different body locations, e.g. left from right humerus or the fourth from the eighth rib. At the same time, it should support a resolution that is high enough to resolve the joint areas, such that adjacent bones are correctly discerned. It should keep the computational burden in a feasible area. For ease of use, it should be trainable end-to-end.

We propose SneakyNet, a multi encoder-decoder network working on inputs of two different resolutions. SneakyNet simultaneously provides high-resolution outputs through its target U-Net, while being served aligned contextual information by the context U-Nets at every level of the encoder-decoder architecture. This end-to-end trainable network can leverage a field of view many times that of the original input while avoiding the cubic growth of intermediate computations. It allows for improved distinct bone segmentation and explicitly reduces the confusion of similar bones in different body locations. Along with our network, we also publish our dataset of 17 upper-body CT scans and the matching manual voxel-wise distinct bone segmentation, including all vertebrae, ribs, and bones of the shoulders, arms, hands, and fingers.

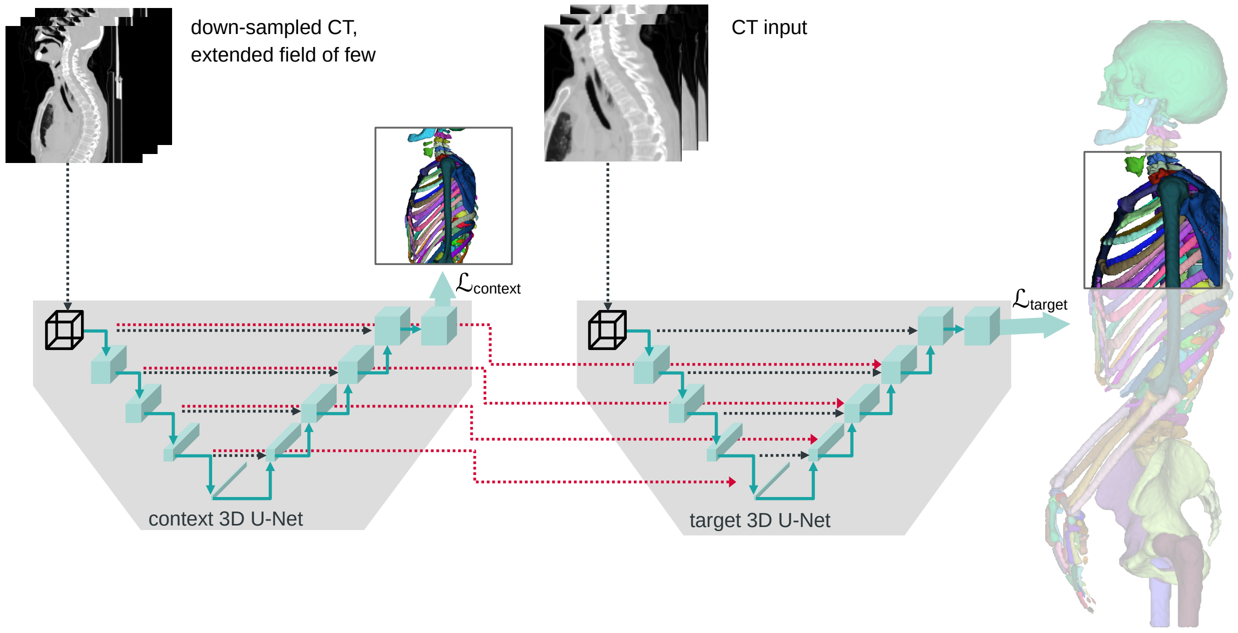


Fig. 1. Task overview: We segment 125 distinct bones from upper-body CT scans using SneakyNet, a multi-encoder-decoder network which incorporates inputs at various resolutions. The example here features one context network, but multiple are possible.

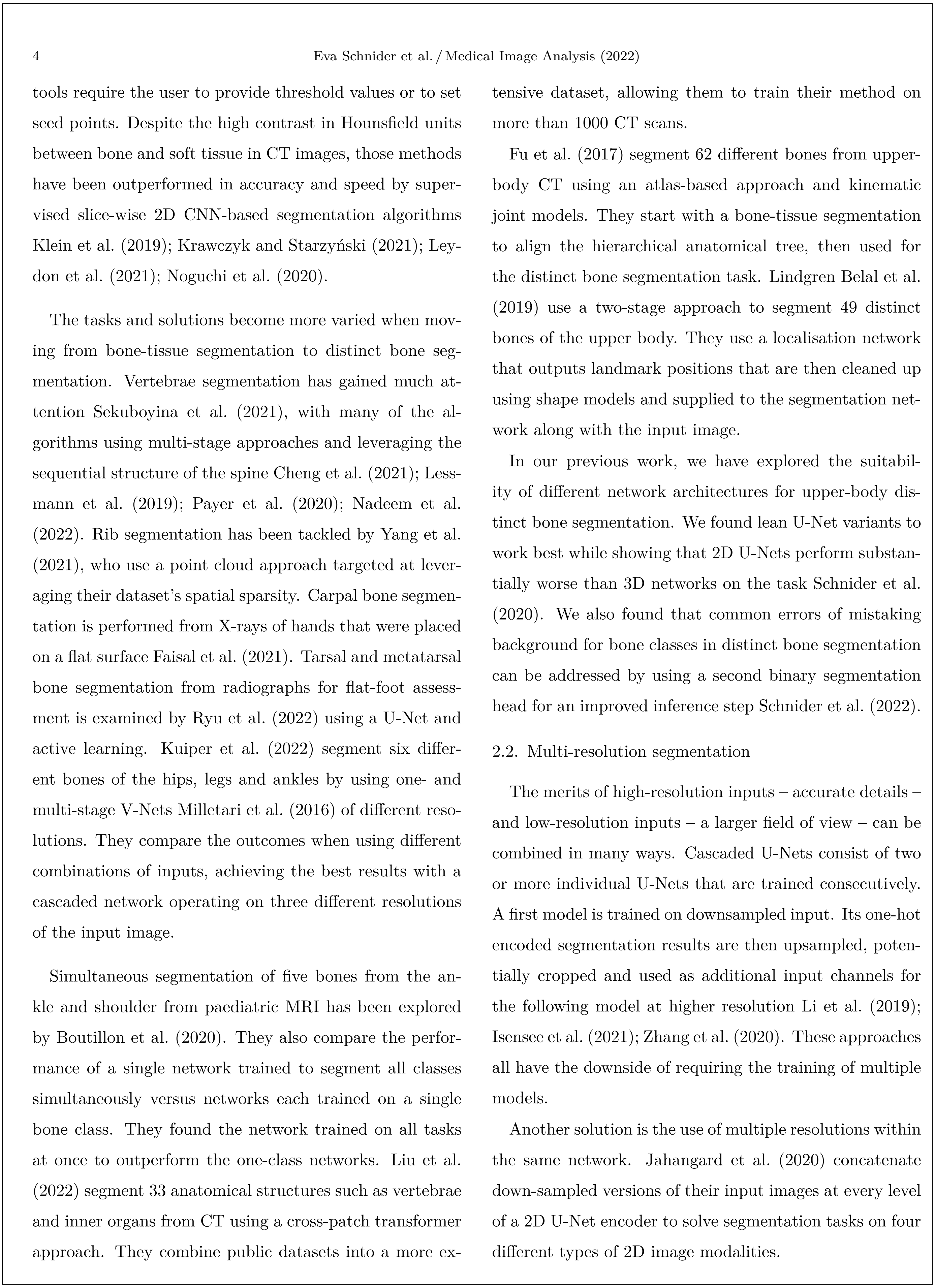
2. Related work

In the last decade, much research has been conducted in medical image segmentation, with CNNs outperforming many previous baseline approaches Kleesiek et al. (2016); Noguchi et al. (2020). Specifically, U-Net-like architectures have been adopted widely in medical image segmentation in both 2D and 3D. The original U-Net Ronneberger et al. (2015) is a 2D encoder-decoder network. Like other fully convolutional segmentation networks Long et al. (2015); Maggiori et al. (2016), its encoder resembles fully-convolutional classification networks and consists of several levels where the spatial resolution is reduced while the number of channels increases. The latent encoding, found at the bottom of the U, is then successively expanded in the spatial directions while the number of channels decreases. Higher and lower resolution information is reconciled through the use of skip-connections that copy information from the encoder to the decoder in the U-Net Ronneberger et al. (2015).

The same ideas can be transferred to three-dimensional inputs to build a 3D U-Net Çiçek et al. (2016), using 3D convolutional kernels. The additional dimension increases the number of trainable parameters and intermediate computations. This increase needs to be countered by a reduced number of channels, a decrease of the batch size, patch-wise sampling, or substitution of up-convolutions by up-sampling Isensee et al. (2019). Isensee et al. (2021) have gone one step further by proposing a self-adapting U-Net framework that chooses architectural configurations empirically and heuristically.

2.1. Distinct Bone segmentation

Bone segmentation from CT usually denotes the differentiation into bone tissue and background. If the bones are further distinguished, we speak of distinct bone segmentation. Bone tissue segmentation from CT can be performed using semi-automated approaches and interactive tools Li and Chen (2021); Fedorov et al. (2012); Yushkevich et al. (2006); Zaimi et al. (2021), which are based on thresholding, morphological operations, region growing, and clustering Argüello et al. (2019); Requist et al. (2021). These



tools require the user to provide threshold values or to set seed points. Despite the high contrast in Hounsfield units between bone and soft tissue in CT images, those methods have been outperformed in accuracy and speed by supervised slice-wise 2D CNN-based segmentation algorithms Klein et al. (2019); Krawczyk and Starzyński (2021); Leydon et al. (2021); Noguchi et al. (2020).

The tasks and solutions become more varied when moving from bone-tissue segmentation to distinct bone segmentation. Vertebrae segmentation has gained much attention Sekuboyina et al. (2021), with many of the algorithms using multi-stage approaches and leveraging the sequential structure of the spine Cheng et al. (2021); Lessmann et al. (2019); Payer et al. (2020); Nadeem et al. (2022). Rib segmentation has been tackled by Yang et al. (2021), who use a point cloud approach targeted at leveraging their dataset's spatial sparsity. Carpal bone segmentation is performed from X-rays of hands that were placed on a flat surface Faisal et al. (2021). Tarsal and metatarsal bone segmentation from radiographs for flat-foot assessment is examined by Ryu et al. (2022) using a U-Net and active learning. Kuiper et al. (2022) segment six different bones of the hips, legs and ankles by using one- and multi-stage V-Nets Milletari et al. (2016) of different resolutions. They compare the outcomes when using different combinations of inputs, achieving the best results with a cascaded network operating on three different resolutions of the input image.

Simultaneous segmentation of five bones from the ankle and shoulder from paediatric MRI has been explored by Boutillon et al. (2020). They also compare the performance of a single network trained to segment all classes simultaneously versus networks each trained on a single bone class. They found the network trained on all tasks at once to outperform the one-class networks. Liu et al. (2022) segment 33 anatomical structures such as vertebrae and inner organs from CT using a cross-patch transformer approach. They combine public datasets into a more extensive dataset, allowing them to train their method on more than 1000 CT scans.

Fu et al. (2017) segment 62 different bones from upper-body CT using an atlas-based approach and kinematic joint models. They start with a bone-tissue segmentation to align the hierarchical anatomical tree, then used for the distinct bone segmentation task. Lindgren Belal et al. (2019) use a two-stage approach to segment 49 distinct bones of the upper body. They use a localisation network that outputs landmark positions that are then cleaned up using shape models and supplied to the segmentation network along with the input image.

In our previous work, we have explored the suitability of different network architectures for upper-body distinct bone segmentation. We found lean U-Net variants to work best while showing that 2D U-Nets perform substantially worse than 3D networks on the task Schnider et al. (2020). We also found that common errors of mistaking background for bone classes in distinct bone segmentation can be addressed by using a second binary segmentation head for an improved inference step Schnider et al. (2022).

2.2. Multi-resolution segmentation

The merits of high-resolution inputs – accurate details – and low-resolution inputs – a larger field of view – can be combined in many ways. Cascaded U-Nets consist of two or more individual U-Nets that are trained consecutively. A first model is trained on downsampled input. Its one-hot encoded segmentation results are then upsampled, potentially cropped and used as additional input channels for the following model at higher resolution Li et al. (2019); Isensee et al. (2021); Zhang et al. (2020). These approaches all have the downside of requiring the training of multiple models.

Another solution is the use of multiple resolutions within the same network. Jahangard et al. (2020) concatenate down-sampled versions of their input images at every level of a 2D U-Net encoder to solve segmentation tasks on four different types of 2D image modalities.

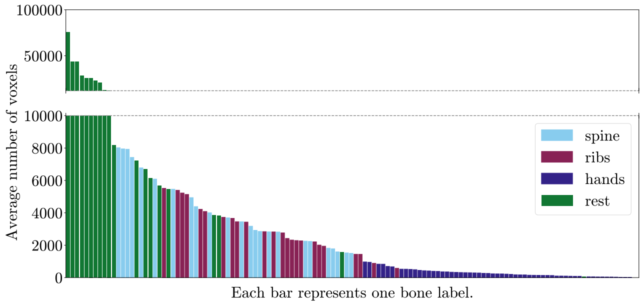


Fig. 2. Size distribution of bones.

Kushnure and Talbar (2021) use multiple resolutions within each U-Net block, where they use varying numbers of convolutions per block on CT slices in liver and tumour segmentation.

Work closest to our own can be found in histopathology whole-slide segmentation. Gu et al. (2018) propose MRN, which uses a 2D target U-Net and one context encoder with drop-skip-connections crossing over at every level. They do not use a context decoder or context loss and work on a binary segmentation problem. Van Rijthoven et al. (2021) propose HookNet, which contains both a target and a context 2D U-Net and two individual losses, such as our work. In contrast to our work, 1) they work with 2D images and 2D networks, while we work in 3D, and 2) they use only one skip connection from the context encoder to the target decoder just before the bottleneck layer, while we use such connections on every level, 3) they use a single context network, while we use up to three. Our experiments show, that our proposed SneakyNet configuration leads to better results on our task, compared to single-resolution networks and to the multi-resolution networks mentioned above.

3. Material and Methods

To assess the performance of SneakyNet, we train it on our in-house upper-body CT dataset, which we provide along with this publication at https://gitlab.com/cian.unibas.ch/sneakynet. We make ablation studies on the combination of context and target information and on the optimal number of context networks.

3.1. Upper-body CT dataset

Table 1. Demographic properties of the full upper-body CT dataset as published. Note that we excluded one scan from the dataset for the computations in this publication because of its unique pose.

	number of volumes	age
Female	8	48-103
Male	9	44-91

The CT images have been acquired post-mortem from body donours by the anatomical department of the University of Basel. A summary of their demographic information is displayed in Table 1. All CT scans were taken with the body donours lying on their backs, and arms placed in front of the body. The arms are bent to various degrees, and the hands overlap in some instances. We omitted one of the scans in our experiments because the arms were folded and the hands crossed that was not seen in any of the other scans.

The manual segmentations have been created using 3D-Slicer Fedorov et al. (2012). Five of the segmentations have been created from scratch on the original resolution of approximately 1 mm. The remaining twelve segmentations have been conducted on scans that were down-sampled to an isotropic resolution of 2 mm with an average size of 237 × 237 × 403 pixels. Instead of creating those segmentations from scratch, an ensemble of networks was used to create an initial automated segmentation which was then manually corrected and refined. Prior to using the scans for training, and for publication we resampled all scans to 2 mm isotropic resolution.

In order to ascertain the anonymity of the body donours, the head area is omitted in the published dataset. We cropped the scan axially just above the first cervical vertebra. The cropping thus affects the bones of two labels: skull and mandible. All other bones are still present in their entirety. The original segmentation comprises one background and 125 foreground labels, one of which we

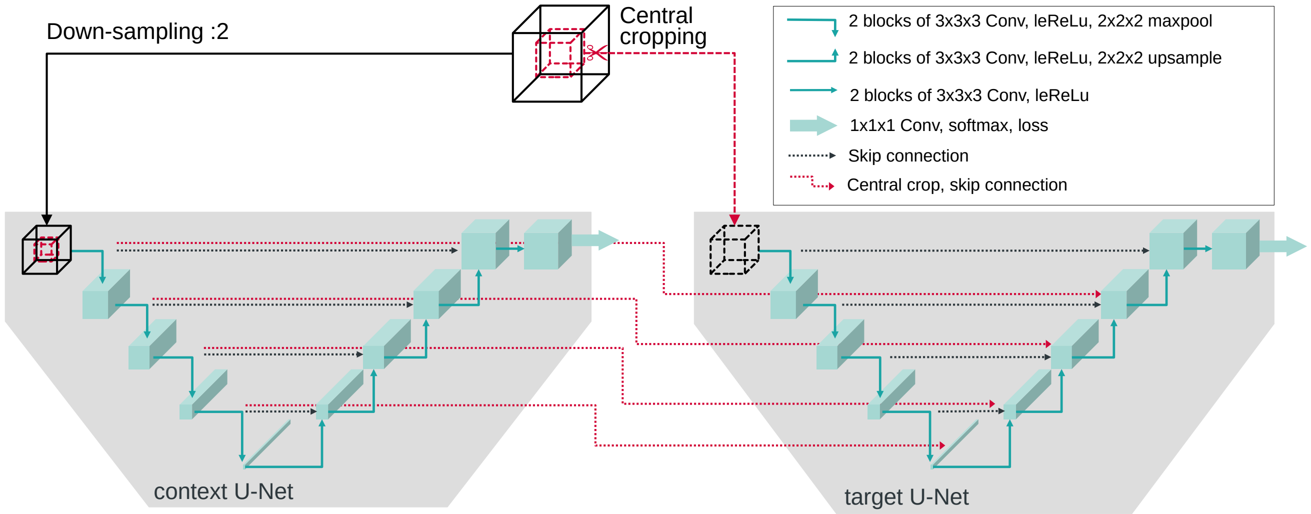

Down-sampling :2
Central cropping
2 blocks of 3x3x3 Conv, leReLu, 2x2x2 maxpool
2 blocks of 3x3x3 Conv, leReLu, 2x2x2 upsample
2 blocks of 3x3x3 Conv, leReLu
1x1x1 Conv, softmax, loss
Skip connection
Central crop, skip connection
context U-Net
target U-Net

Fig. 3. Overview of the architecture for one target and one context network. Left: the context 3D U-Net is working on low-resolution data with a larger field of view. At each encoder level, two skip connections leave the final convolutional layer. The conventional skip connections link the encoder to the decoder, while the crop-skip connections concatenate the central cropped content with a deeper level of the target decoder. Right: The target 3D U-Net working with high-resolution data operates only on the central part of the original input. The target decoder receives skip connections both from its own encoder and the context encoder. Both networks have their individual classification head and loss functions and work on the same voxel size input patches.

omit in the published dataset since it is a pacemaker, not a type of bone.

There is a substantial imbalance in the number of pixels per label, not only between background- and foreground labels but also among foreground labels. While certain large bones feature 40,000 pixels and more, most of the bones of the hand comprise 1,000 pixels and less, see Figure 2.

The dataset can be accessed at https://gitlab.com/cian.unibas.ch/sneakynet.

3.2. SneakyNet Architecture

We present a visual overview of the architecture with one context network in Figure 3. In general, however, SneakyNet consists of one target network and one or more context networks. The target network operates on high-resolution data and eventually produces the desired segmentation maps. The context networks operate on lower resolution inputs spanning a larger field of view. Information is propagated from the context networks to the target network using crop-skip connections presented in Section 3.2.1.

On their own, the individual context and target networks follow a lean variant Isensee et al. (2019) of the 3D U-Net Çiçek et al. (2016), which uses same-padding and simple upsampling instead of upconvolutions. Furthermore, we use instance normalization instead of batch normalization since the computational requirements force us to work with a batch size of 1. In our baseline computations, where we have only a target network and omit the context networks, we use twice as many channels as proposed in the original publication Isensee et al. (2019) in order for our variants and the baselines to have approximately the same number of trainable parameters. Inputs to the network are required to be multiples of 2^{M-1}, where M denotes the number of levels of the U-Net. We use the basic architecture of $M = 5$ and therefore need multiples of 16 pixels in every dimension as input.

For the target network we use inputs of size (Sx, Sy, Sz) at full resolution. For each of the context networks we use that input plus its surrounding area, which together span a field of view of $2^{\kappa} \cdot (Sx, Sy, Sz)$. We display the case of $\kappa = 1$ in Figure 3, but also use context networks with $\kappa = 2$

Table 2. Comparison of architectures with different field of view (FOV) of their target and context network(s). in terms of trainable parameters, time, and memory requirements.

Config	Target network FOV	Context network(s) FOV	trainable parameters	Nr. of input pixels	training time per iteration (s)
A 3D U-Net	32^3	—	$5.8 \cdot 10^7$	$3.3 \cdot 10^4$	0.44
	64^3	—		$26.2 \cdot 10^4$	0.57
3D U-Net slim*	128^3	—	$1.5 \cdot 10^7$	$209.7 \cdot 10^4$	4.24
B Hook Net	32^3	64^3	$3.7 \cdot 10^7$	$6.6 \cdot 10^4$	0.41
	64^3	128^3		$52.4 \cdot 10^4$	0.72
C MRN	32^3	64^3	$4.7 \cdot 10^7$	$6.6 \cdot 10^4$	0.43
	64^3	128^3		$52.4 \cdot 10^4$	1.27
D SneakyNet (ours), $\mathcal{L}_{\text{X-Ent}}$ only	32^3	64^3	$4.9 \cdot 10^7$	$6.6 \cdot 10^4$	0.41
	64^3	128^3		$52.4 \cdot 10^4$	0.86
D SneakyNet (ours)	32^3	64^3	$4.9 \cdot 10^7$	$6.6 \cdot 10^4$	0.45
		$64^3 - 128^3$	$5.8 \cdot 10^7$	$9.9 \cdot 10^4$	0.70
		$64^3 - 128^3 - 256^3$	$6.2 \cdot 10^7$	$13.1 \cdot 10^4$	3.16
	64^3	128^3	$4.9 \cdot 10^7$	$52.4 \cdot 10^4$	1.28
		$128^3 - 256^3$	$5.8 \cdot 10^7$	$78.6 \cdot 10^4$	3.11

* Operating the full 3D U-Net on patches of size 128^3 exceeds the available GPU memory.

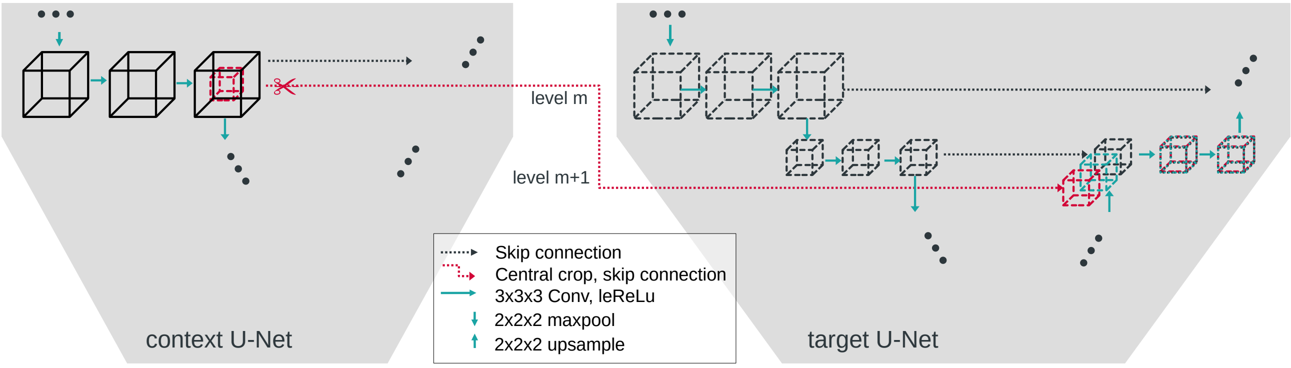


Fig. 4. Detailed view of the architecture. Displayed are only two out of five levels of the U-Nets. Left: the context U-Net working on low-resolution data with a larger field of view. Right: The U-Net working with the central cropped high-resolution data. After all encoder convolutions of level m, a cropped copy of the output is skipped to the target decoder at level $m + 1$. The decoder receives skip connections from its own encoder and the context network. The intermediate results of the decoder and both skip connections are concatenated along the channel axis before undergoing further convolutions.

and $\kappa = 3$ in our ablation studies. The context network inputs are down-sampled to reduce their size to (Sx, Sy, Sz). We perform the down-sampling using $(2^\kappa \times 2^\kappa \times 2^\kappa)$ average-pooling with a stride of 2^κ. Both target and context network inputs eventually have a size of (Sx, Sy, Sz), but at different resolutions and fields of view.

3.2.1. Crop-skip connections

We use crop-skip connections to transfer information from the context to the target branch. We crop the encoder output at the desired level m such that only the centre cube of half the size per dimension remains. This centre cube is now spatially aligned to the input of the tar-

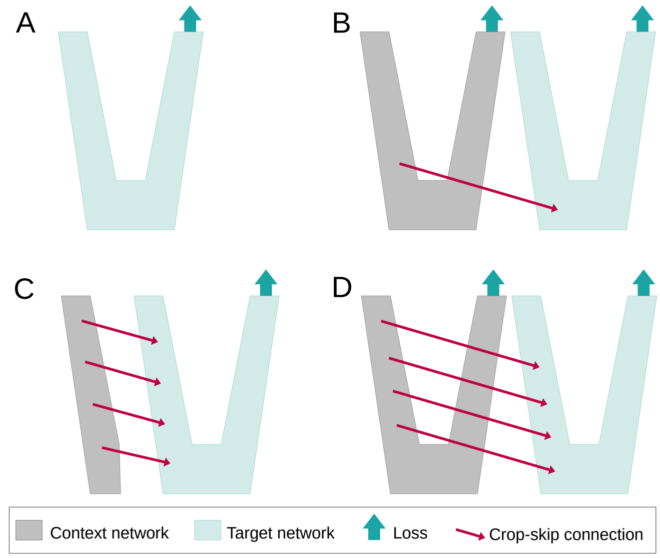


Fig. 5. Schematic of the four network configurations used in our ablation study. A shows a base U-Net, while B, C, D show different possibilities of how to insert information into the target network, see also Section 3.2.1 for a written description.

get branch. We concatenate the centre cube to the next lower level $m + 1$ of the target decoder to match the spatial size. We refer to the central cropping and subsequent concatenation into a lower level of the target branch as crop-skip-connection. A detailed schematic of the crop-skip connection is depicted in Figure 4.

We explore three network configurations which differ in their number of crop-skip connections and their use of a context loss, and compare it to a baseline U-Net. A visual comparison of the architectures is given in Figure 5 and the parameters are provided in Table 2.

- A – Baseline: Lean variant of a 3D U-Net.
- B – Hook Net: One context network with a single crop-skip connection is added to the target network. The crop-skip connection enters the target network at its bottleneck layer. This configuration is used in Van Rijthoven et al. (2021).
- C – MRN: Crop-skip connections connect the context encoder and the target decoder at every level. There is neither a context decoder nor a context loss function. This configuration was used in Gu et al. (2018).
- D – proposed: Crop-skip connections connect all levels of the context and target networks. The context network has a decoder with its own loss function.

3.3. Training

Our dataset (see also Table 1) is split into 11 scans for training, 2 for validation and 3 for testing. We use 5-fold cross-validation, ensuring that every scan appears in precisely one of the cross-validation folds in the test set.

The loss is composed of an unweighted combination of the target network's loss and the losses of the K context networks.

$$\mathcal{L}_{\text{total}} := \mathcal{L}_{\text{target}} + \sum_{\kappa=1}^{K} \mathcal{L}^{\kappa}_{\text{context}} \qquad (1)$$

For both networks, we combine two loss functions, i.e. the cross-entropy loss $\mathcal{L}_{\text{X-Ent}}$ and the Dice-Loss $\mathcal{L}_{\text{DSC}}$ Milletari et al. (2016).

$$\mathcal{L}_{[\text{target}|\text{context}]} := \mathcal{L}_{\text{X-Ent}} + \sum_{c \in C} \mathcal{L}^{c}_{\text{DSC}} \qquad (2)$$

The Dice-Loss itself is an unweighted sum over the Dice-Loss for each individual class c.

We optimized the network weights using the Adam optimizer Kingma and Ba (2014) with an initial learning rate of 0.001. We trained our networks for 100000 iterations until convergence was observed.

Our input images are padded by $(S - S_{\text{target}})/2$ all-around using edge value padding. The padding step ensures that we can sample high-resolution patch centres right to the image's border.

We implemented and trained our networks using Tensorflow Keras 2.5. All training and inference were conducted on NVidia Quadro RTX 6000 GPUs of 24 GB RAM size.

3.4. Evaluation

We evaluate the performance of our models using the Dice Score Coefficient (DSC). For an individual foreground class c, the DSC is defined as a function of true positives TP_c, false positives FP_c and false negatives FN_c of that class:

$$\text{DSC}_c = \frac{2\text{TP}_c}{2\text{TP}_c + \text{FP}_c + \text{FN}_c}, \qquad (3)$$

To indicate the performance over all classes, we give the median and the 16 and 84 quantiles (1σ) over all classes c. To not give a distorted impression of the distribution, we exclude classes where $\mathrm{TP_c} = 0$ and therefore $\mathrm{DSC_c} = 0$. We present the percentage of classes included in brackets in Table 3 and Table 4 to make up for the omission.

4. Results and Discussion

Our experiments show how automated distinct bone segmentation can be improved using a SneakyNet based multi-resolution approach. We evaluate our results on multiple target resolutions with different numbers of context networks and field of view sizes and perform an ablation study to determine the most beneficial way to combine context and target network information.

We evaluated some of the most common errors when using a baseline segmentation method. We found that the missing context information leads to similar-looking bones in different body regions being mistaken for one another. In the confusion matrix presented in Figure 6, we observe that when using a baseline 3D U-Net, humerus pixels were predicted as femur, and the left and right humerus were confused for one another (right confusion matrix). When using context information, these errors are reduced almost entirely (left confusion matrix).

We performed an ablation study to see how different strategies of combining the context and target information within the network perform. In Table 3 we present the quantitative results. For both target patch sizes, 32 and 64, all strategies (B-D) improve upon the baseline 3D U-Net (A). The observed effect is substantially bigger when using the smaller target patch size of 32^3, where the median DCS rises from 0.64 to 0.75. The DSC still increases from 0.83 to 0.86 median DSC on the bigger target patches.

The combination of skip connections at every level and a context loss function in our proposed architecture increases the accuracy further, as compared to the Hook Net Van Rijthoven et al. (2021) and the MRN Gu et al. (2018). Cross-entropy only or cross-entropy plus Dice loss leads to very similar results on both patch sizes. To ensure that the improvements do not stem from the increased number of trainable parameters alone, we set the number of convolutional channels in the baseline 3D U-Net such that its number of trainable parameters matches or supersedes that of the variants, see Table 2.

In Table 4 and Figure 7 we compare the influence of different numbers of context networks. Qualitative results are depicted in Figure 8. We go up to an input size of 256^3 pixels for the context branches (before down-sampling), which leads to a maximum of three context networks for a target patch size of 32^3, and two context networks for target patches of size 64^3. The best results were achieved in both cases when using context patches with a field of view of up to 128^3 pixels. For the target patches of 32^3 pixels, the median DSC raises from 0.75 for one context network to 0.79 for two. Adding another context network with an even bigger field of view very slightly increases the fraction of nonzero DSC classes but does not affect the median DSC. While using the baseline 3D U-Net on patches of size 32^3 yields very noisy results, the addition of two context networks allows for segmentations close in accuracy to those conducted on patches of size 64^3, while using less than half as many input pixels and the same number of trainable parameters. Using more than one context branch on the input patches of size 64^3 does not further improve the segmentation results. We assume that for our dataset using the current 2 mm resolution all context information needed to assign bone classes correctly is contained in a field of view of 128^3 pixels, and thus, bigger input regions are not necessary in this case. For higher resolution data or different segmentation tasks more context levels or larger input sizes might be needed.

5. Conclusion

We propose SneakyNet for improved distinct bone segmentation from CT. 3D CNNs suffer from cubic growth of

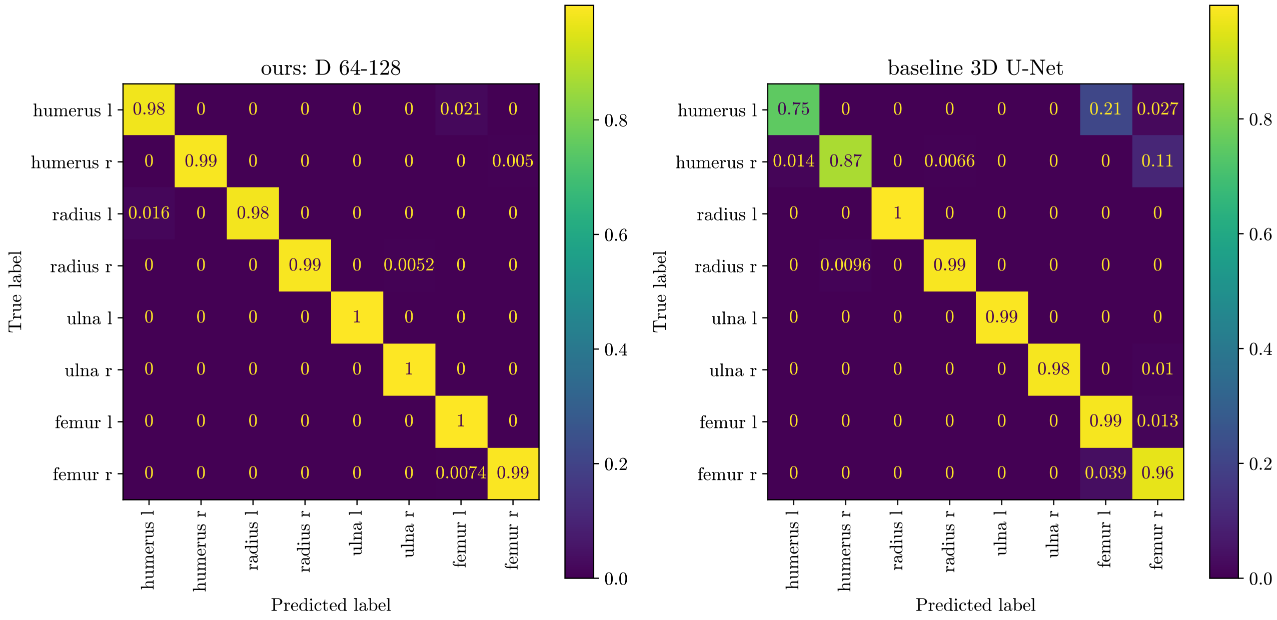


Fig. 6. Confusion matrix among the long bones of the arms and legs. With our method, there is considerably less confusion between the left and right sides of the body and between arm and leg bones.

Table 3. Ablation results in DSC for different model configurations.

Target patch size	32				64			
DSC	Median	σ	$-\sigma$	non-zero DSC	Median	σ	$-\sigma$	non-zero DS
A baseline 3D U-Net	0.64	+0.19	-0.34	94.5%	0.83	+0.09	-0.27	94.5
B Hook Net	0.66	+0.17	-0.34	94.1%	0.85	+0.09	-0.32	95.3
C MRN	0.69	+0.16	-0.37	95.1%	0.84	+0.09	-0.31	96.0
D SneakyNet (ours), $\mathcal{L}_{\text{X-Ent}}$ only	0.73	+0.15	-0.32	95.1%	0.86	+0.09	-0.27	96.7
D SneakyNet (ours)	0.75	+0.14	-0.33	95.3%	0.86	+0.08	-0.28	96.7

their intermediate computations when using larger input patches. In practice, this means that only patches of comparably modest size can be used as inputs for 3D CNNs. At the same time, the resolution of medical images is ever increasing, and thus the field of view of a fixed number of pixels is decreasing.

We, therefore, propose a network architecture that uses additional inputs at a lower resolution but with a larger field of view to provide the necessary context information to assign the proper bone classes. We compared three different ways of combining the context and target information and evaluated the results using zero to three context networks. Using context networks improves the segmentation results on all target patch sizes.

We make our code and a de-identified version of our dataset of 17 upper-body CT scans with voxel-wise CT bone labelling publicly available.

Acknowledgements

This work was financially supported by the Werner Siemens Foundation through the MIRACLE project. The segmentation of the dataset where conducted by [add acknowledgement for Anatomy here].

References

Argüello, D., Acevedo, H.G.S., González-Estrada, O.A., 2019. Comparison of segmentation tools for structural analysis of bone tis-

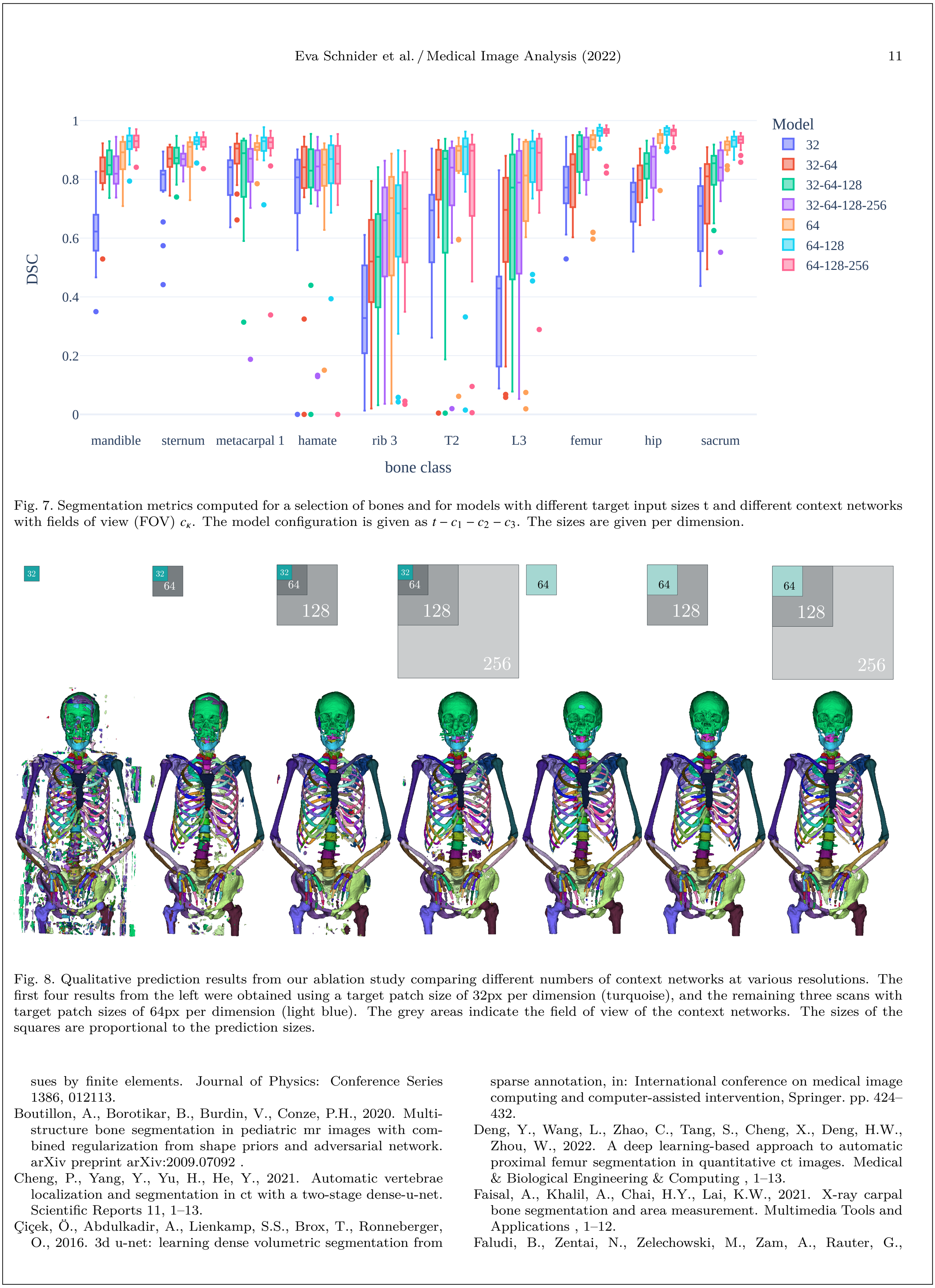


Fig. 7. Segmentation metrics computed for a selection of bones and for models with different target input sizes t and different context networks with fields of view (FOV) c_κ. The model configuration is given as $t - c_1 - c_2 - c_3$. The sizes are given per dimension.

Fig. 8. Qualitative prediction results from our ablation study comparing different numbers of context networks at various resolutions. The first four results from the left were obtained using a target patch size of 32px per dimension (turquoise), and the remaining three scans with target patch sizes of 64px per dimension (light blue). The grey areas indicate the field of view of the context networks. The sizes of the squares are proportional to the prediction sizes.

sues by finite elements. Journal of Physics: Conference Series 1386, 012113.

Boutillon, A., Borotikar, B., Burdin, V., Conze, P.H., 2020. Multi-structure bone segmentation in pediatric mr images with combined regularization from shape priors and adversarial network. arXiv preprint arXiv:2009.07092 .

Cheng, P., Yang, Y., Yu, H., He, Y., 2021. Automatic vertebrae localization and segmentation in ct with a two-stage dense-u-net. Scientific Reports 11, 1–13.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3d u-net: learning dense volumetric segmentation from sparse annotation, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 424–432.

Deng, Y., Wang, L., Zhao, C., Tang, S., Cheng, X., Deng, H.W., Zhou, W., 2022. A deep learning-based approach to automatic proximal femur segmentation in quantitative ct images. Medical & Biological Engineering & Computing , 1–13.

Faisal, A., Khalil, A., Chai, H.Y., Lai, K.W., 2021. X-ray carpal bone segmentation and area measurement. Multimedia Tools and Applications , 1–12.

Faludi, B., Zentai, N., Zelechowski, M., Zam, A., Rauter, G.,

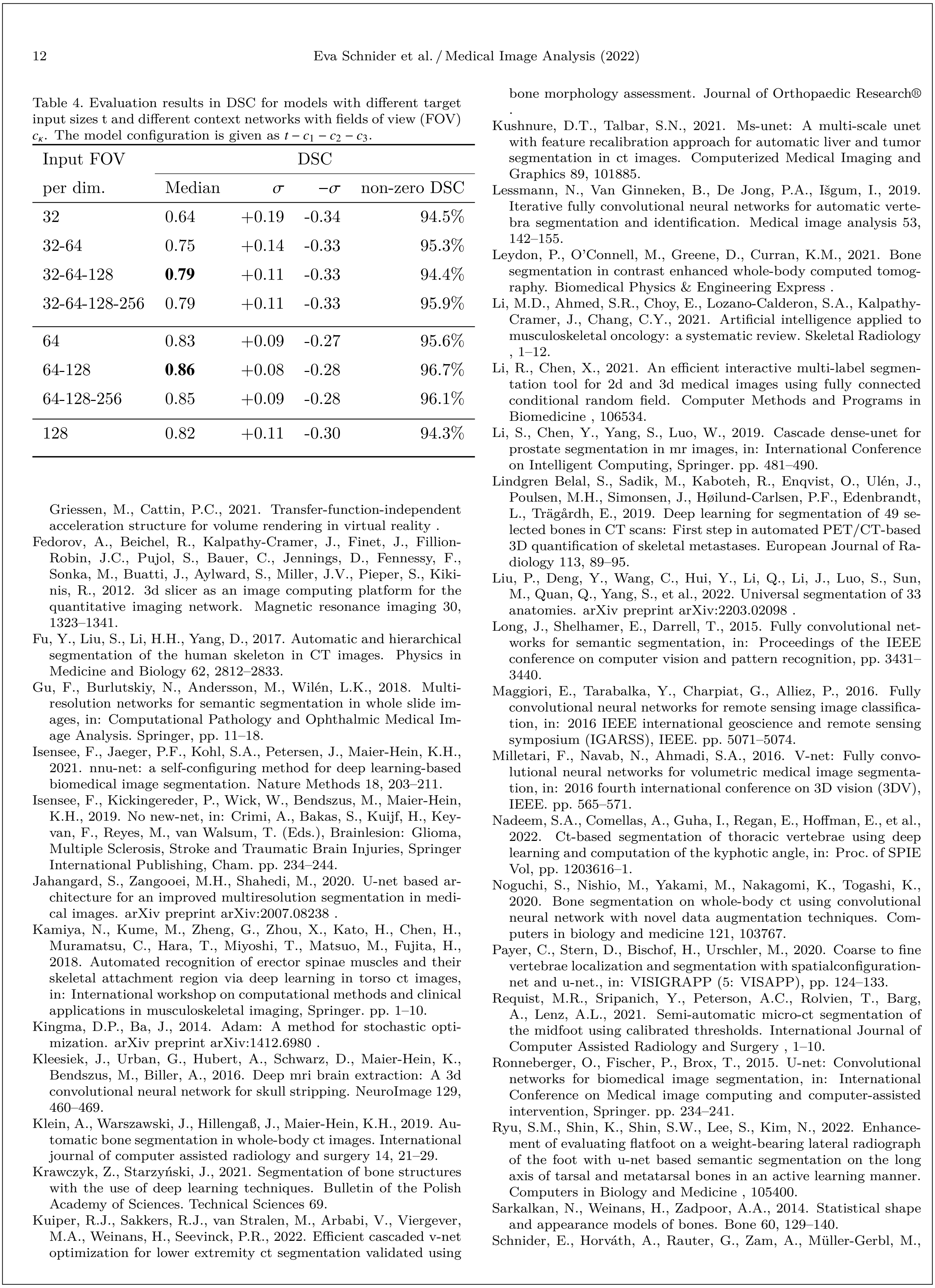


Table 4. Evaluation results in DSC for models with different target input sizes t and different context networks with fields of view (FOV) c_k. The model configuration is given as $t - c_1 - c_2 - c_3$.

Input FOV	DSC			
per dim.	Median	σ	$-\sigma$	non-zero DSC
32	0.64	+0.19	-0.34	94.5%
32-64	0.75	+0.14	-0.33	95.3%
32-64-128	**0.79**	+0.11	-0.33	94.4%
32-64-128-256	0.79	+0.11	-0.33	95.9%
64	0.83	+0.09	-0.27	95.6%
64-128	**0.86**	+0.08	-0.28	96.7%
64-128-256	0.85	+0.09	-0.28	96.1%
128	0.82	+0.11	-0.30	94.3%

Griessen, M., Cattin, P.C., 2021. Transfer-function-independent acceleration structure for volume rendering in virtual reality .

Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.C., Pujol, S., Bauer, C., Jennings, D., Fennessy, F., Sonka, M., Buatti, J., Aylward, S., Miller, J.V., Pieper, S., Kikinis, R., 2012. 3d slicer as an image computing platform for the quantitative imaging network. Magnetic resonance imaging 30, 1323–1341.

Fu, Y., Liu, S., Li, H.H., Yang, D., 2017. Automatic and hierarchical segmentation of the human skeleton in CT images. Physics in Medicine and Biology 62, 2812–2833.

Gu, F., Burlutskiy, N., Andersson, M., Wilén, L.K., 2018. Multi-resolution networks for semantic segmentation in whole slide images, in: Computational Pathology and Ophthalmic Medical Image Analysis. Springer, pp. 11–18.

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

Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., Maier-Hein, K.H., 2019. No new-net, in: Crimi, A., Bakas, S., Kuijf, H., Keyvan, F., Reyes, M., van Walsum, T. (Eds.), Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, Springer International Publishing, Cham. pp. 234–244.

Jahangard, S., Zangooei, M.H., Shahedi, M., 2020. U-net based architecture for an improved multiresolution segmentation in medical images. arXiv preprint arXiv:2007.08238 .

Kamiya, N., Kume, M., Zheng, G., Zhou, X., Kato, H., Chen, H., Muramatsu, C., Hara, T., Miyoshi, T., Matsuo, M., Fujita, H., 2018. Automated recognition of erector spinae muscles and their skeletal attachment region via deep learning in torso ct images, in: International workshop on computational methods and clinical applications in musculoskeletal imaging, Springer. pp. 1–10.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Kleesiek, J., Urban, G., Hubert, A., Schwarz, D., Maier-Hein, K., Bendszus, M., Biller, A., 2016. Deep mri brain extraction: A 3d convolutional neural network for skull stripping. NeuroImage 129, 460–469.

Klein, A., Warszawski, J., Hillengaß, J., Maier-Hein, K.H., 2019. Automatic bone segmentation in whole-body ct images. International journal of computer assisted radiology and surgery 14, 21–29.

Krawczyk, Z., Starzyński, J., 2021. Segmentation of bone structures with the use of deep learning techniques. Bulletin of the Polish Academy of Sciences. Technical Sciences 69.

Kuiper, R.J., Sakkers, R.J., van Stralen, M., Arbabi, V., Viergever, M.A., Weinans, H., Seevinck, P.R., 2022. Efficient cascaded v-net optimization for lower extremity ct segmentation validated using bone morphology assessment. Journal of Orthopaedic Research® .

Kushnure, D.T., Talbar, S.N., 2021. Ms-unet: A multi-scale unet with feature recalibration approach for automatic liver and tumor segmentation in ct images. Computerized Medical Imaging and Graphics 89, 101885.

Lessmann, N., Van Ginneken, B., De Jong, P.A., Išgum, I., 2019. Iterative fully convolutional neural networks for automatic vertebra segmentation and identification. Medical image analysis 53, 142–155.

Leydon, P., O'Connell, M., Greene, D., Curran, K.M., 2021. Bone segmentation in contrast enhanced whole-body computed tomography. Biomedical Physics & Engineering Express .

Li, M.D., Ahmed, S.R., Choy, E., Lozano-Calderon, S.A., Kalpathy-Cramer, J., Chang, C.Y., 2021. Artificial intelligence applied to musculoskeletal oncology: a systematic review. Skeletal Radiology , 1–12.

Li, R., Chen, X., 2021. An efficient interactive multi-label segmentation tool for 2d and 3d medical images using fully connected conditional random field. Computer Methods and Programs in Biomedicine , 106534.

Li, S., Chen, Y., Yang, S., Luo, W., 2019. Cascade dense-unet for prostate segmentation in mr images, in: International Conference on Intelligent Computing, Springer. pp. 481–490.

Lindgren Belal, S., Sadik, M., Kaboteh, R., Enqvist, O., Ulén, J., Poulsen, M.H., Simonsen, J., Høilund-Carlsen, P.F., Edenbrandt, L., Trägårdh, E., 2019. Deep learning for segmentation of 49 selected bones in CT scans: First step in automated PET/CT-based 3D quantification of skeletal metastases. European Journal of Radiology 113, 89–95.

Liu, P., Deng, Y., Wang, C., Hui, Y., Li, Q., Li, J., Luo, S., Sun, M., Quan, Q., Yang, S., et al., 2022. Universal segmentation of 33 anatomies. arXiv preprint arXiv:2203.02098 .

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440.

Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P., 2016. Fully convolutional neural networks for remote sensing image classification, in: 2016 IEEE international geoscience and remote sensing symposium (IGARSS), IEEE. pp. 5071–5074.

Milletari, F., Navab, N., Ahmadi, S.A., 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 fourth international conference on 3D vision (3DV), IEEE. pp. 565–571.

Nadeem, S.A., Comellas, A., Guha, I., Regan, E., Hoffman, E., et al., 2022. Ct-based segmentation of thoracic vertebrae using deep learning and computation of the kyphotic angle, in: Proc. of SPIE Vol, pp. 1203616–1.

Noguchi, S., Nishio, M., Yakami, M., Nakagomi, K., Togashi, K., 2020. Bone segmentation on whole-body ct using convolutional neural network with novel data augmentation techniques. Computers in biology and medicine 121, 103767.

Payer, C., Stern, D., Bischof, H., Urschler, M., 2020. Coarse to fine vertebrae localization and segmentation with spatialconfiguration-net and u-net., in: VISIGRAPP (5: VISAPP), pp. 124–133.

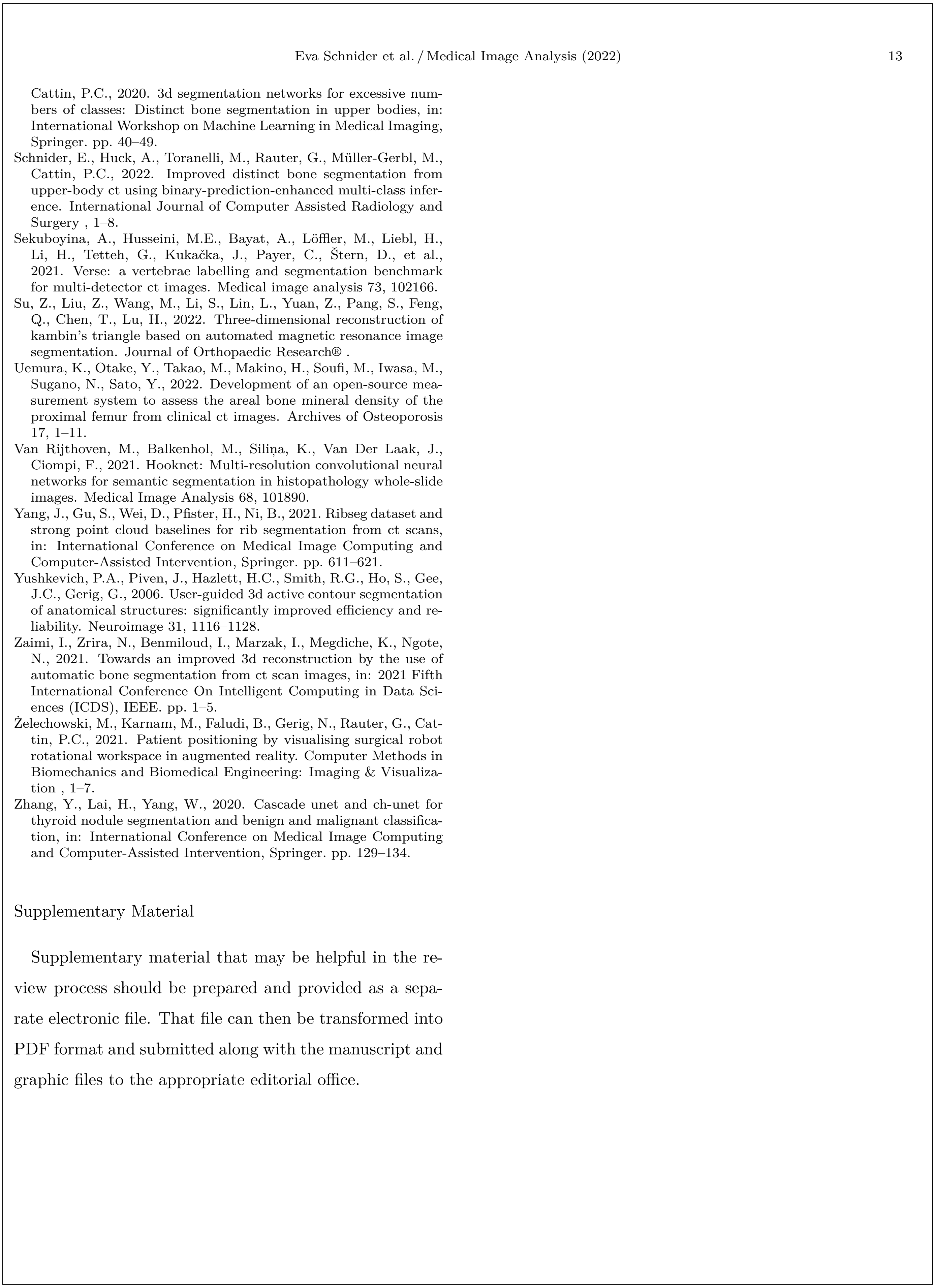
Requist, M.R., Sripanich, Y., Peterson, A.C., Rolvien, T., Barg, A., Lenz, A.L., 2021. Semi-automatic micro-ct segmentation of the midfoot using calibrated thresholds. International Journal of Computer Assisted Radiology and Surgery , 1–10.

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

Ryu, S.M., Shin, K., Shin, S.W., Lee, S., Kim, N., 2022. Enhancement of evaluating flatfoot on a weight-bearing lateral radiograph of the foot with u-net based semantic segmentation on the long axis of tarsal and metatarsal bones in an active learning manner. Computers in Biology and Medicine , 105400.

Sarkalkan, N., Weinans, H., Zadpoor, A.A., 2014. Statistical shape and appearance models of bones. Bone 60, 129–140.

Schnider, E., Horváth, A., Rauter, G., Zam, A., Müller-Gerbl, M.,



Cattin, P.C., 2020. 3d segmentation networks for excessive numbers of classes: Distinct bone segmentation in upper bodies, in: International Workshop on Machine Learning in Medical Imaging, Springer. pp. 40–49.

Schnider, E., Huck, A., Toranelli, M., Rauter, G., Müller-Gerbl, M., Cattin, P.C., 2022. Improved distinct bone segmentation from upper-body ct using binary-prediction-enhanced multi-class inference. International Journal of Computer Assisted Radiology and Surgery , 1–8.

Sekuboyina, A., Husseini, M.E., Bayat, A., Löffler, M., Liebl, H., Li, H., Tetteh, G., Kukačka, J., Payer, C., Štern, D., et al., 2021. Verse: a vertebrae labelling and segmentation benchmark for multi-detector ct images. Medical image analysis 73, 102166.

Su, Z., Liu, Z., Wang, M., Li, S., Lin, L., Yuan, Z., Pang, S., Feng, Q., Chen, T., Lu, H., 2022. Three-dimensional reconstruction of kambin's triangle based on automated magnetic resonance image segmentation. Journal of Orthopaedic Research® .

Uemura, K., Otake, Y., Takao, M., Makino, H., Soufi, M., Iwasa, M., Sugano, N., Sato, Y., 2022. Development of an open-source measurement system to assess the areal bone mineral density of the proximal femur from clinical ct images. Archives of Osteoporosis 17, 1–11.

Van Rijthoven, M., Balkenhol, M., Siliņa, K., Van Der Laak, J., Ciompi, F., 2021. Hooknet: Multi-resolution convolutional neural networks for semantic segmentation in histopathology whole-slide images. Medical Image Analysis 68, 101890.

Yang, J., Gu, S., Wei, D., Pfister, H., Ni, B., 2021. Ribseg dataset and strong point cloud baselines for rib segmentation from ct scans, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 611–621.

Yushkevich, P.A., Piven, J., Hazlett, H.C., Smith, R.G., Ho, S., Gee, J.C., Gerig, G., 2006. User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability. Neuroimage 31, 1116–1128.

Zaimi, I., Zrira, N., Benmiloud, I., Marzak, I., Megdiche, K., Ngote, N., 2021. Towards an improved 3d reconstruction by the use of automatic bone segmentation from ct scan images, in: 2021 Fifth International Conference On Intelligent Computing in Data Sciences (ICDS), IEEE. pp. 1–5.

Żelechowski, M., Karnam, M., Faludi, B., Gerig, N., Rauter, G., Cattin, P.C., 2021. Patient positioning by visualising surgical robot rotational workspace in augmented reality. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization , 1–7.

Zhang, Y., Lai, H., Yang, W., 2020. Cascade unet and ch-unet for thyroid nodule segmentation and benign and malignant classification, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 129–134.

Supplementary Material

Supplementary material that may be helpful in the review process should be prepared and provided as a separate electronic file. That file can then be transformed into PDF format and submitted along with the manuscript and graphic files to the appropriate editorial office.

Chapter 9

Discussion and Conclusion

Is it possible to simultaneously segment all bones of a human body from a CT scan using deep learning? This was the question at the onset of this Ph.D. project. In the following sections, we will summarise our accomplishments, discuss the limitations of our work, and suggest how to proceed in future work.

9.1 Contributions

In this work, we presented different methods to segment distinct bones from upper-body CT scans. Distinct bone segmentation from CT is a difficult task due to the presence of over 200 different bones in the human body and the large size of the CT scans used. In our first segmentation method we have provided a proof of concept that fully supervised end-to-end trained U-Net-like networks are capable of multi-class bone segmentation of upper-body CT scans. Compared to other published approaches, we have increased the maximum number of distinct bones segmented in one method from 88 [37], (respectively 49 [73] for neural network-based approaches) to 125.

To remedy the most prevalent errors occurring during our distinct bone segmentation task, we proposed inference modifications that are potentially useful to a wide range of 3D multi-class segmentation problems: Our proposed BEM inference is suitable for any multi-class segmentation problem and can help to improve foreground and background distinction. It splits the task into two sub-problems: bone and non-bone distinction and the identification of individual bones. We showed that the approach works with dual-decoder networks specifically designed for the task but also when using a binary bone/non-bone segmentation that has been obtained by other means. Since the labels of any multi-class task can be easily transformed into binary background/foreground labels, this approach can be retrofitted to any segmentation network.

To tackle the misclassification of similar looking bones in different locations of the bone, we designed SneakyNet, a multi-resolution network. It incorporates increasingly large fields of view with decreasing resolutions into one network and thus includes more spatial context while keeping the number of input pixels manageable. This is particularly useful when we want to work with higher resolution data, where the currently computationally manageable input sizes (in terms of pixels) capture too little area of the CT scan.

In collaboration with our group's VR and AR team, we integrated the trained models for distinct bone segmentation into their SpectoVR application, where it can be used to segment new CT scans on the fly. The time needed to generate the segmentation is 1-5 minutes, depending on the scan size and model chosen. The resulting segmentation is then directly visualised and editable in VR. It can be used for any upstream task, including the envisioned planning and navigation software for the MIRACLE project.

9.2 Limitations

Naturally, there are numerous limitations to our work. They can be roughly divided into two groups: limitations in model generalisation originating from the data we used to train our models and limitations of the approaches themselves irrespective of the training data.

Generalisation: We used data provided by the University's anatomical institute. Those CT scans were taken post-mortem of body donors. This demographic is very specific in many ways: It mainly consists of elderly people, some over 100 years old, with frequent occurrence of mild forms of scoliosis and calcification quite typical for old age. Another issue is the pose: To decrease the radiation dose, living persons are usually scanned in a different pose when conducting an upper body CT scan. They lift their arms above their heads and out of the scanner's field-of-view. The tissue properties are potentially slightly affected by post-mortem decay at the scan-time. Thus the trained model might not be perfect for living subjects. In addition, all of our data came from the same CT scanner. While different CT scanners produce somewhat similar outputs, in contrast to MRI, scanner invariance of the trained model is not guaranteed.

The dataset was quite varied in terms of sex and body build, but not so in age: Only CT scans of adults were used in our work. We do not anticipate our trained algorithms to generalise to children's CT because not all ossification centres are fused in children, and their bodies have different proportions.

There were no individuals with supernumerary bones in our dataset and none with large metal implants and their accompanying streak artefacts. Furthermore, we did not observe pathologic or extreme variations in bone shape and layout, such as very pronounced scoliosis or bone cancer, in our dataset. Therefore, we do not expect our trained models to be robust against those modes of variation.

Our initial dataset comprised an upper and a lower body scan of five donors. There were no full-body scans available due to hardware limitations. We focused our investigations on the upper-body dataset for two primary reasons: It contained more individual bones from more groups (see Section 2.1), making the task more interesting. It also contained some very frequently segmented bones, i.e., the vertebrae from which we hoped to get additional data for pretraining.

Concluding, we expect our trained models at this stage to work for upper-body CT scans of human adults with a standard bone anatomy and without metal implants or strong bone deformations, irrespective of their build and sex. We expect better performance for older adults and post-mortem scans. Extending the model's capabilities to other anatomies or demographics should be possible using an appropriate training dataset.

Model performance:

We achieved DSC values of up to 0.85 (median over all bones), meaning that the algorithm usually performed decently on most bones. Certain bones, however, often stayed undetected or were wrongly labelled. This behaviour was most frequently observed in the hands where parts of finger bones were misidentified. Another location of frequent misclassification was the spine, where the separation between vertebrae was not always satisfactory, and the labels tended to be off by one vertebra. This behaviour has been improved using SneakyNet but still deserves further consideration.

9.3 Future Work

Our final datasets contained only 17 scans, due to the very time-intensive labelling process. We see great potential in extending the dataset to improve the results further and increase the robustness of the trained models.

We have investigated active-learning strategies to select the next CT scans that should be labelled and included into the training dataset. We followed the hypothesis that scans that incur higher ensemble uncertainty scores during inference time might provide more new information to the model and lead to better results when included in the dataset. In our experiments we did not observe such an effect, which warrants further investigations into the best policy to decide on what new data to label and include.

Given the difficulty in obtaining voxel-wise labelled data, weakly or unsupervised approaches may be a viable option. This could take the form of a generative model with the task of creating a specific bone, or possibly many bones at once, and a discriminator with the task of identifying real from computer-generated bone, similar to the detection of brain tumours [138]. Such weakly supervised approaches still require labels, typically containing the knowledge of whether a specific volume of the scan contains the class of interest in place of voxel-wise segmentation. Although a bounding box or rough delineation of the bones is still necessary, this may require less work than a voxel-wise approach. With some bones being very close to one another, the question remains, how much error would be introduced by a bounding box approach and how this would affect the outcome.

We use the multi-resolution SneakyNet approach to provide additional spatial information to our models. Another option would be to directly provide spatial information using an encoding similar to the one used in transformer networks. We tried such approaches, but they are infamously data-hungry and we did not yet obtain acceptable results. Nevertheless, transformers are still being researched very actively and adapted to new and smaller datasets, which commends a revisit.

Finally, it may be interesting to create segmentations that are very accurate by using articulated atlases, shape models, or generative human models. A hierarchical approach would likely be necessary, with the trunk being fitted first and the extremities coming next, joint by joint. As a result, the majority of the segmentation task would change to a registration problem, which has its own set of challenges. These approaches would, however, potentially lose the advantage of being trainable end-to-end and risk ending up with tedious tinkering. Even if registration-based methods may end up being more accurate, the automated deep-learning segmentation developed in this thesis could still be used as a first step to guarantee a good initial alignment, which is typically a requirement for a successful registration outcome.

9.4 Conclusion

In this work, we propose a deep learning-based method for automated distinct bone segmentation in upper-body CT scans. Our method is currently used in the SpectoVR software to segment bones on-the-fly. We found a working method based on 3D U-Nets that forms our baseline and subsequently identified the main modes of errors and proposed additions to deal with those errors. Our number of simultaneously segmented bones is higher than previously published in the literature. The inference of an upper-body CT scan takes approximately 3 minutes, depending on the input size, and input window overlap. This makes our algorithm suitable for visualisation and as a prior for other more fine-grained algorithms. In terms of accuracy, our methods are best suited to tasks, where the general location of multiple bones is more critical than the perfect segmentation of any single bone. If the segmentation of one specific single bone is desired, more specialised algorithms might be better suited to the task. We propose to use our method in gross anatomy education, for navigation within a user interface that displays CT scans or as an intermediate step in pose computation.

Bibliography

[1] Paul Aljabar, Rolf A Heckemann, Alexander Hammers, Joseph V Hajnal, and Daniel Rueckert. Multi-atlas based segmentation of brain images: atlas selection and its effect on accuracy. *Neuroimage*, 46(3):726–738, 2009.

[2] Simon Andermatt, Antal Horváth, Simon Pezold, and Philippe Cattin. Pathology segmentation using distributional differences to images of healthy origin. In *International MICCAI Brainlesion Workshop*, pages 228–238. Springer, 2018.

[3] Andrea Apicella, Francesco Donnarumma, Francesco Isgrò, and Roberto Prevete. A survey on modern trainable activation functions. *Neural Networks*, 138:14–32, 2021.

[4] Arthur Appel. Some techniques for shading machine renderings of solids. In *Proceedings of the April 30–May 2, 1968, spring joint computer conference*, pages 37–45, 1968.

[5] Hossein Arabi and Habib Zaidi. Comparison of atlas-based techniques for whole-body bone segmentation. *Medical image analysis*, 36:98–112, 2017.

[6] André Araujo, Wade Norris, and Jack Sim. Computing receptive fields of convolutional neural networks. *Distill*, 2019. doi: 10.23915/distill.00021. https://distill.pub/2019/computing-receptive-fields.

[7] D Argüello, H G Sánchez Acevedo, and O A González-Estrada. Comparison of segmentation tools for structural analysis of bone tissues by finite elements. *Journal of Physics: Conference Series*, 1386:012113, nov 2019.

[8] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[9] Hyun-Jin Bae, Heejung Hyun, Younghwa Byeon, Keewon Shin, Yongwon Cho, Young Ji Song, Seong Yi, Sung-Uk Kuh, Jin S Yeom, and Namkug Kim. Fully automated 3d segmentation and separation of multiple cervical vertebrae in ct images using a 2d convolutional neural network. *Computer Methods and Programs in Biomedicine*, 184:105119, 2020.

[10] Miguel Angel Gonzalez Ballester, Andrew P Zisserman, and Michael Brady. Estimation of the partial volume effect in mri. *Medical image analysis*, 6(4):389–405, 2002.

[11] Marie Bieth, Loic Peter, Stephan G Nekolla, Matthias Eiber, Georg Langs, Markus Schwaiger, and Bjoern Menze. Segmentation of skeleton and organs in whole-body ct images via iterative trilateration. *IEEE transactions on medical imaging*, 36(11):2276–2286, 2017.

[12] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[13] Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. *Advances in neural information processing systems*, 20, 2007.

[14] Yuri Y Boykov and M-P Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, volume 1, pages 105–112. IEEE, 2001.

[15] Wael Brahim, Makram Mestiri, Nacim Betrouni, and Kamel Hamrouni. Semi-automated rib cage segmentation in ct images for mesothelioma detection. In *2016 International Image Processing, Applications and Systems (IPAS)*, pages 1–6. IEEE, 2016.

[16] Samuel Budd, Emma C Robinson, and Bernhard Kainz. A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical Image Analysis*, 71:102062, 2021.

[17] Helen R Buie, Graeme M Campbell, R Joshua Klinck, Joshua A MacNeil, and Steven K Boyd. Automatic segmentation of cortical and trabecular compartments based on a dual threshold technique for in vivo micro-ct bone analysis. *Bone*, 41(4):505–515, 2007.

[18] David B. Burr. Bone Morphology and Organization. In *Basic and Applied Bone Biology (Second Edition)*, pages 3–26. Academic Press, Cambridge, MA, USA, Jan 2019. ISBN 978-0-12-813259-3. doi: 10.1016/B978-0-12-813259-3.00001-4.

[19] Jerrold T Bushberg and John M Boone. *The essential physics of medical imaging*. Lippincott Williams & Wilkins, 2011.

[20] Thorsten Buzug. *Computed Tomography*. Springer, Berlin, Germany, 2008. doi: 10.1007/978-3-540-39408-2.

[21] Eduardo Castro, Jaime S Cardoso, and Jose Costa Pereira. Elastic deformations for data augmentation in breast cancer mass detection. In *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pages 230–234. IEEE, 2018.

[22] Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.

[23] Richard Anthony Crowther, DJ DeRosier, and Aaron Klug. The reconstruction of a three-dimensional structure from projections and its application to electron microscopy. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 317 (1530):319–340, 1970.

[24] William R Crum, Oscar Camara, and Derek LG Hill. Generalized overlap measures for evaluation and validation in medical image analysis. *IEEE transactions on medical imaging*, 25(11):1451–1461, 2006.

[25] Sanjoy Dasgupta. Analysis of a greedy active learning strategy. *Advances in neural information processing systems*, 17, 2004.

[26] Michal Drozdzal, Eugene Vorontsov, Gabriel Chartrand, Samuel Kadoury, and Chris Pal. The importance of skip connections in biomedical image segmentation. In *Deep learning and data labeling for medical applications*, pages 179–187. Springer, 2016.

[27] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016.

[28] Albrecht Dürer. *Underweysung der Messung, mit dem Zirckel und Richtscheyt, in Linien, Ebenen unnd gantzen corporen*. Hieronymus Andreae, Nüremberg, 1525. URL http://digital.slub-dresden.de/werkansicht/dlf/17139/5/.

[29] LC Ebert, A Dobay, S Franckenberg, MJ Thali, S Decker, and J Ford. Image segmentation of post-mortem computed tomography data in forensic imaging: Methods and applications. *Forensic Imaging*, page 200483, 2021.

[30] Balázs Faludi, Norbert Zentai, Marek Zelechowski, Azhar Zam, Georg Rauter, Mathias Griessen, and Philippe C. Cattin. Transfer-Function-Independent Acceleration Structure for Volume Rendering in Virtual Reality. In *High-Performance Graphics - Symposium Papers*. The Eurographics Association, 2021. doi: 10.2312/hpg.20211279.

[31] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic resonance imaging*, 30(9):1323–1341, 2012.

[32] Javier Ferrer-Torregrosa, Miguel Ángel Jiménez-Rodríguez, Javier Torralba-Estelles, Fernanda Garzón-Farinós, Marcelo Pérez-Bermejo, and Nadia Fernández-Ehrling. Distance learning ects and flipped classroom in the anatomy learning: comparative study of the use of augmented reality, video and notes. *BMC medical education*, 16(1):1–9, 2016.

[33] Lucas Fidon, Wenqi Li, Luis C Garcia-Peraza-Herrera, Jinendra Ekanayake, Neil Kitchen, Sébastien Ourselin, and Tom Vercauteren. Generalised wasserstein dice score for imbalanced multi-class segmentation using holistic convolutional networks. In *International MICCAI brainlesion workshop*, pages 64–76. Springer, 2017.

[34] Martin Fiebich, Christopher M Straus, Vivek Sehgal, Bernhard C Renger, Kunio Doi, and Kenneth R Hoffmann. Automatic bone segmentation technique for ct angiographic studies. *Journal of computer assisted tomography*, 23(1):155–161, 1999.

[35] Jung-Leng Foo, Marisol Martinez-Escobar, Bethany Juhnke, Keely Cassidy, Kenneth Hisley, Thom Lobe, and Eliot Winer. Evaluating mental workload of two-dimensional and three-dimensional visualization for anatomical structure localization. *Journal of Laparoendoscopic & Advanced Surgical Techniques*, 23(1):65–70, 2013.

[36] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

[37] Yabo Fu, Shi Liu, H Harold Li, and Deshan Yang. Automatic and hierarchical segmentation of the human skeleton in CT images. *Physics in Medicine and Biology*, 62(7): 2812–2833, April 2017.

[38] P Furnstahl, T Fuchs, Andreas Schweizer, Ladislav Nagy, Gábor Székely, and Matthias Harders. Automatic and robust forearm segmentation using graph cuts. In *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 77–80. IEEE, 2008.

[39] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[40] C Glide-Hurst, D Chen, H Zhong, and IJ Chetty. Changes realized from extended bit-depth and metal artifact reduction in ct. *Medical physics*, 40(6Part1):061711, 2013.

[41] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[42] Edwin R. Hancock and Josef Kittler. Edge-labeling using dictionary-based relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):165–181, 1990.

[43] Thomas N Hangartner. Thresholding technique for accurate analysis of density and geometry in qct, pqct and muct images. *Journal of Musculoskeletal and Neuronal Interactions*, 7(1):9, 2007.

[44] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3d medical image segmentation: a review. *Medical image analysis*, 13(4):543–563, 2009.

[45] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

[46] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[47] Daniel P Huttenlocher, Gregory A. Klanderman, and William J Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 1993.

[48] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[49] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2):203–211, 2021.

[50] Ivana Isgum, Marius Staring, Annemarieke Rutten, Mathias Prokop, Max A Viergever, and Bram Van Ginneken. Multi-atlas-based segmentation with local decision fusion—application to cardiac and aortic segmentation in ct scans. *IEEE transactions on medical imaging*, 28(7):1000–1010, 2009.

[51] Rens Janssens, Guodong Zeng, and Guoyan Zheng. Fully automatic segmentation of lumbar vertebrae from ct images using cascaded 3d fully convolutional networks. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 893–897. IEEE, 2018.

[52] Zhanghexuan Ji, Yan Shen, Chunwei Ma, and Mingchen Gao. Scribble-based hierarchical weakly supervised learning for brain tumor segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 175–183. Springer, 2019.

[53] Pamela T Johnson, David G Heath, Donald F Bliss, Brian Cabral, and Elliot K Fishman. Three-dimensional ct: real-time interactive volume rendering. *AJR. American journal of roentgenology*, 167(3):581–583, 1996.

[54] Maria Kallergi, Kevin Woods, Laurence P Clarke, Wei Qian, and Robert A Clark. Image segmentation in digital mammography: comparison of local thresholding and region growing algorithms. *Computerized medical imaging and graphics*, 16(5):323–331, 1992.

[55] Naoki Kamiya, Masanori Kume, Guoyan Zheng, Xiangrong Zhou, Hiroki Kato, Huayue Chen, Chisako Muramatsu, Takeshi Hara, Toshiharu Miyoshi, Masayuki Matsuo, et al. Automated recognition of erector spinae muscles and their skeletal attachment region via deep learning in torso ct images. In *International workshop on computational methods and clinical applications in musculoskeletal imaging*, pages 1–10. Springer, 2018.

[56] Yan Kang, Klaus Engelke, and Willi A Kalender. A new accurate and precise 3-d segmentation method for skeletal structures in volumetric ct data. *IEEE transactions on medical imaging*, 22(5):586–598, 2003.

[57] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[58] Henry J Kelley. Gradient theory of optimal flight paths. *Ars Journal*, 30(10):947–954, 1960.

[59] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[60] André Klein, Jan Warszawski, Jens Hillengaß, and Klaus H Maier-Hein. Automatic bone segmentation in whole-body ct images. *International journal of computer assisted radiology and surgery*, 14(1):21–29, 2019.

[61] Ronald Kline. Cybernetics, automata studies, and the dartmouth conference on artificial intelligence. *IEEE Annals of the History of Computing*, 33(4):5–16, 2011. doi: 10.1109/MAHC.2010.44.

[62] Marcel Krčah, Gábor Székely, and Rémi Blanc. Fully automatic and fast segmentation of the femur bone from 3d-ct images with no shape prior. In *2011 IEEE international symposium on biomedical imaging: from nano to macro*, pages 2087–2090. IEEE, 2011.

[63] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[64] Daniela Kugelmann, Leonard Stratmann, Nils Nühlen, Felix Bork, Saskia Hoffmann, Golbarg Samarbarksh, Anna Pferschy, Anna Maria Von der Heide, Andreas Eimannsberger, Pascal Fallavollita, et al. An augmented reality magic mirror as additive teaching device for gross anatomy. *Annals of Anatomy-Anatomischer Anzeiger*, 215:71–77, 2018.

[65] Yann LeCun, D Touresky, G Hinton, and T Sejnowski. A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pages 21–28, 1988.

[66] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[67] Yann LeCun et al. Generalization and network design strategies. *Connectionism in perspective*, 19(143-155):18, 1989.

[68] Nikolas Lessmann, Bram Van Ginneken, Pim A De Jong, and Ivana Išgum. Iterative fully convolutional neural networks for automatic vertebra segmentation and identification. *Medical image analysis*, 53:142–155, 2019.

[69] Marc Levoy. Efficient ray tracing of volume data. *ACM Transactions on Graphics (TOG)*, 9(3):245–261, 1990.

[70] Patrick Leydon, Martin O'Connell, Derek Greene, and Kathleen M Curran. Bone segmentation in contrast enhanced whole-body computed tomography. *Biomedical Physics & Engineering Express*, 8(5):055010, 2022.

[71] Ruizhe Li and Xin Chen. An efficient interactive multi-label segmentation tool for 2d and 3d medical images using fully connected conditional random field. *Computer Methods and Programs in Biomedicine*, page 106534, 2021.

[72] Charlene Liew. The future of radiology augmented with artificial intelligence: a strategy for success. *European journal of radiology*, 102:152–156, 2018.

[73] Sarah Lindgren Belal, May Sadik, Reza Kaboteh, Olof Enqvist, Johannes Ulén, Mads H. Poulsen, Jane Simonsen, Poul F. Høilund-Carlsen, Lars Edenbrandt, and Elin Trägårdh. Deep learning for segmentation of 49 selected bones in CT scans: First step in automated PET/CT-based 3D quantification of skeletal metastases. *European Journal of Radiology*, 113:89–95, April 2019. ISSN 0720048X.

[74] Seppo Linnainmaa. *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors*. PhD thesis, Master's Thesis, Univ. Helsinki, 1970.

[75] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[76] Shuang Liu, Yiting Xie, and Anthony P Reeves. Individual bone structure segmentation and labeling from low-dose chest ct. In *Medical imaging 2017: computer-aided diagnosis*, volume 10134, pages 1063–1073. SPIE, 2017.

[77] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[78] Le Lu, Dijia Wu, Nathan Lay, David Liu, Isabella Nogues, and Ronald M Summers. Accurate 3d bone segmentation in challenging ct images: Bottom-up parsing and contextualized optimization. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10. IEEE, 2016.

[79] Meng Ma, Pascal Fallavollita, Ina Seelbach, Anna Maria Von Der Heide, Ekkehard Euler, Jens Waschke, and Nassir Navab. Personalized augmented reality for anatomy education. *Clinical Anatomy*, 29(4):446–453, 2016.

[80] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.

[81] Jordi Minnema, Maureen van Eijnatten, Wouter Kouw, Faruk Diblen, Adriënne Mendrik, and Jan Wolff. Ct image segmentation of bone for medical additive manufacturing using a convolutional neural network. *Computers in biology and medicine*, 103:130–139, 2018.

[82] Marvin Minsky and Seymour Papert. An introduction to computational geometry. *Cambridge tiass., HIT*, 479:480, 1969.

[83] Francisco J Moreno-Barea, Fiammetta Strazzera, José M Jerez, Daniel Urda, and Leonardo Franco. Forward noise adjustment scheme for data augmentation. In *2018 IEEE symposium series on computational intelligence (SSCI)*, pages 728–734. IEEE, 2018.

[84] Elise F Morgan and Louis C Gerstenfeld. The bone organ system: form and function. In *Marcus and Feldman's Osteoporosis*, pages 15–35. Elsevier, 2021.

[85] Christian Moro, Zane Štromberga, Athanasios Raikos, and Allan Stirling. The effectiveness of virtual and augmented reality in health sciences and medical anatomy. *Anatomical sciences education*, 10(6):549–559, 2017.

[86] Magdalena Müller-Gerbl, Reinhard Putz, Norbert Hodapp, Erik Schulte, and Berthold Wimmer. Computed tomography-osteoabsorptiometry for assessing the density distribution of subchondral bone as a measure of long-term mechanical adaptation in individual joints. *Skeletal radiology*, 18(7):507–512, 1989.

[87] Yasuo Nakajima, Kei Yamada, Keiko Imamura, and Kazuko Kobayashi. Radiologist supply and workload: international comparison. *Radiation medicine*, 26(8):455–465, 2008.

[88] Shunjiro Noguchi, Mizuho Nishio, Masahiro Yakami, Keita Nakagomi, and Kaori Togashi. Bone segmentation on whole-body ct using convolutional neural network with novel data augmentation techniques. *Computers in biology and medicine*, 121:103767, 2020.

[89] Alireza Norouzi, Mohd Shafry Mohd Rahim, Ayman Altameem, Tanzila Saba, Abdolvahab Ehsani Rad, Amjad Rehman, and Mueen Uddin. Medical image segmentation methods, algorithms, and applications. *IETE Technical Review*, 31(3):199–213, 2014.

[90] OECD. Computed tomography (ct) scanners (indicator). https://doi.org/10.1787/bedece12-en (Accessed on 14 July 2022), 2022.

[91] OECD. Computed tomography (ct) exams (indicator). https://doi.org/10.1787/3c994537-en (Accessed on 14 July 2022), 2022.

[92] OECD. Magnetic resonance imaging (mri) units (indicator). https://doi.org/10.1787/1a72e7d1-en (Accessed on 20 July 2022), 2022.

[93] OECD. Magnetic resonance imaging (mri) exams (indicator). https://doi.org/10.1787/1d89353f-en (Accessed on 20 July 2022), 2022.

[94] Mohammadreza Asghari Oskoei and Huosheng Hu. A survey on edge detection methods. *University of Essex, UK*, 33, 2010.

[95] Nigel Palastanga, Derek Field, and Roger Soames. Components of the musculoskeletal system. In *Anatomy and Human Movement*, pages 12–27. Butterworth-Heinemann, Oxford, England, UK, Jan 1989. ISBN 978-0-433-00032-7. doi: 10.1016/B978-0-433-00032-7.50006-1.

[96] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[97] José-Antonio Pérez-Carrasco, Begoña Acha, Cristina Suárez-Mejías, Jose-Luis López-Guerra, and Carmen Serrano. Joint segmentation of bones and muscles using an intensity and histogram-based energy minimization approach. *Computer methods and programs in biomedicine*, 156:85–95, 2018.

[98] Dzung L Pham, Chenyang Xu, and Jerry L Prince. A survey of current methods in medical image segmentation. *Annual review of biomedical engineering*, 2(3):315–337, 2000.

[99] Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

[100] Bingjiang Qiu, Jiapan Guo, Joep Kraeima, Haye H Glas, Ronald JH Borra, Max JH Witjes, and Peter MA van Ooijen. Automatic segmentation of the mandible from computed tomography scans for 3d virtual surgical planning using the convolutional neural network. *Physics in Medicine & Biology*, 64(17):175020, 2019.

[101] Dana Rahbani, Andreas Morel-Forster, Dennis Madsen, Marcel Lüthi, and Thomas Vetter. Robust registration of statistical shape models for unsupervised pathology annotation. In *Large-Scale Annotation of Biomedical Data and Expert Label Synthesis and Hardware Aware Learning for Medical Imaging and Computer Assisted Intervention*, pages 13–21. Springer, 2019.

[102] Kanchana Rathnayaka, Tony Sahama, Michael A Schuetz, and Beat Schmutz. Effects of ct image segmentation methods on the accuracy of long bone 3d reconstructions. *Medical engineering & physics*, 33(2):226–233, 2011.

[103] Georg Rauter. The miracle. In *Lasers in Oral and Maxillofacial Surgery*, pages 247–253. Springer, 2020.

[104] Melissa R Requist, Yantarat Sripanich, Andrew C Peterson, Tim Rolvien, Alexej Barg, and Amy L Lenz. Semi-automatic micro-ct segmentation of the midfoot using calibrated thresholds. *International Journal of Computer Assisted Radiology and Surgery*, pages 1–10, 2021.

[105] Abi Rimmer. Radiologist shortage leaves patient care at risk, warns royal college. *BMJ: British Medical Journal (Online)*, 359, 2017.

[106] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[107] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[108] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[109] Nazli Sarkalkan, Harrie Weinans, and Amir A Zadpoor. Statistical shape and appearance models of bones. *Bone*, 60:129–140, 2014.

[110] Howard I Scher, Michael J Morris, Walter M Stadler, Celestia Higano, Ethan Basch, Karim Fizazi, Emmanuel S Antonarakis, Tomasz M Beer, Michael A Carducci, Kim N Chi, et al. Trial design and objectives for castration-resistant prostate cancer: updated recommendations from the prostate cancer clinical trials working group 3. *Journal of Clinical Oncology*, 34(12):1402, 2016.

[111] Eva Schnider, Antal Horváth, Georg Rauter, Azhar Zam, Magdalena Müller-Gerbl, and Philippe C Cattin. 3d segmentation networks for excessive numbers of classes: Distinct bone segmentation in upper bodies. In *International Workshop on Machine Learning in Medical Imaging*, pages 40–49. Springer, 2020.

[112] Eva Schnider, Antal Huck, Mireille Toranelli, Georg Rauter, Magdalena Müller-Gerbl, and Philippe C Cattin. Improved distinct bone segmentation from upper-body ct using binary-prediction-enhanced multi-class inference. *International Journal of Computer Assisted Radiology and Surgery*, pages 1–8, 2022.

[113] Eva Schnider, Antal Huck, Mireille Toranelli, Georg Rauter, Azhar Zam, Magdalena Müller-Gerbl, and Philippe Cattin. Ensemble uncertainty as a criterion for dataset expansion in distinct bone segmentation from upper-body ct images. *arXiv*, 2022. doi: 10.48550/ARXIV.2208.09216. URL https://arxiv.org/abs/2208.09216.

[114] Michael Schuenke, Erik Schulte, Udo Schumacher, Lawrence M Ross, Edward D Lamperti, and Markus Voll. *Thieme atlas of anatom0:y general anatomy and musculoskeletal system*. THIEME Atlas of Anatomy. Thieme, 1st edition edition, 2010.

[115] Heiko Seim, Dagmar Kainmueller, Markus Heller, Hans Lamecker, Stefan Zachow, and Hans-Christian Hege. Automatic segmentation of the pelvic bones from ct data based on a statistical shape model. *VCBM*, 8:93–100, 2008.

[116] Anjany Sekuboyina, Markus Rempfler, Jan Kukačka, Giles Tetteh, Alexander Valentinitsch, Jan S Kirschke, and Bjoern H Menze. Btrfly net: Vertebrae labelling with energy-based adversarial learning of local spine prior. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 649–657. Springer, 2018.

[117] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1aIuk-RW.

[118] Neal E Seymour, Anthony G Gallagher, Sanziana A Roman, Michael K O'brien, Vipin K Bansal, Dana K Andersen, and Richard M Satava. Virtual reality training improves operating room performance: results of a randomized, double-blinded study. *Annals of surgery*, 236(4):458, 2002.

[119] Neeraj Sharma and Lalit M Aggarwal. Automated medical image segmentation techniques. *Journal of medical physics/Association of Medical Physicists of India*, 35(1):3, 2010.

[120] Sagar Sharma, Simone Sharma, and Anidhya Athaiya. Activation functions in neural networks. *towards data science*, 6(12):310–316, 2017.

[121] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[122] Nathan Sjoquist. *A novel approach for the visualisation and progression tracking of metastatic bone disease*. PhD thesis, University of Cambridge, 2021.

[123] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[124] Sebastian M Staubli, Peter Maloca, Christoph Kuemmerli, Julia Kunz, Amanda S Dirnberger, Andreas Allemann, Julian Gehweiler, Savas Soysal, Raoul Droeser, Silvio Däster, Gabriel Hess, Dimitri Raptis, Otto Kollmar, Markus von Flüe, and Philippe Cattin. Magnetic resonance cholangiopancreatography enhanced by virtual reality as a novel tool to improve the understanding of biliary anatomy and the teaching of surgical trainees. *Frontiers in Surgery*, 9, 2022.

[125] D Gentry Steele and Claud A Bramblett. *The anatomy and biology of the human skeleton*. Texas A&M University Press, 1988.

[126] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien 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*, pages 240–248. Springer, 2017.

[127] Elham Taghizadeh, Alexandre Terrier, Fabio Becce, Alain Farron, and Philippe Büchler. Automated ct bone segmentation using statistical shape modelling and local template matching. *Computer methods in biomechanics and biomedical engineering*, 22(16): 1303–1310, 2019.

[128] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):1–28, 2015.

[129] Luke Taylor and Geoff Nitschke. Improving deep learning with generic data augmentation. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1542–1547. IEEE, 2018.

[130] Neslisah Torosdagli, Denise K Liberton, Payal Verma, Murat Sincan, Janice Lee, Sumanta Pattanaik, and Ulas Bagci. Robust and fully automated segmentation of mandible from ct scans. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 1209–1212. IEEE, 2017.

[131] A. M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950. ISSN 00264423, 14602113. URL http://www.jstor.org/stable/2251299.

[132] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[133] Mark JJP Van Grinsven, Bram van Ginneken, Carel B Hoyng, Thomas Theelen, and Clara I Sánchez. Fast convolutional neural network training using selective data sampling: Application to hemorrhage detection in color fundus images. *IEEE transactions on medical imaging*, 35(5):1273–1284, 2016.

[134] M Viceconti, R Lattanzi, B Antonietti, S Paderni, R Olmi, A Sudanese, and A Toni. Ct-based surgical planning software improves the accuracy of total hip replacement preoperative planning. *Medical engineering & physics*, 25(5):371–377, 2003.

[135] Daniel Weiskopf. *GPU-based interactive visualization techniques*. Springer, 2007.

[136] C-F Westin, S Warfield, Abhir Bhalerao, L Mui, J Richolt, and Ron Kikinis. Tensor controlled local structure enhancement of ct images for bone segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 1205–1212. Springer, 1998.

[137] Tim D White and Pieter A Folkens. *Human osteology*. Gulf Professional Publishing, 2000.

[138] Julia Wolleb, Robin Sandkühler, and Philippe C Cattin. Descargan: Disease-specific anomaly detection with weak supervision. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 14–24. Springer, 2020.

[139] David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[140] Dijia Wu, David Liu, Zoltan Puskas, Chao Lu, Andreas Wimmer, Christian Tietjen, Grzegorz Soza, and S Kevin 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*, pages 980–987. IEEE, 2012.

[141] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[142] Jiancheng Yang, Shixuan Gu, Donglai Wei, Hanspeter Pfister, and Bingbing Ni. Ribseg dataset and strong point cloud baselines for rib segmentation from ct scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 611–621. Springer, 2021.

[143] Paul A Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C Gee, and Guido Gerig. User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *Neuroimage*, 31 (3):1116–1128, 2006.

[144] Imane Zaimi, Nabila Zrira, Ibtissam Benmiloud, Imad Marzak, Kawtar Megdiche, and Nabil Ngote. Towards an improved 3d reconstruction by the use of automatic bone segmentation from ct scan images. In *2021 Fifth International Conference On Intelligent Computing in Data Sciences (ICDS)*, pages 1–5. IEEE, 2021.

[145] Marek Żelechowski, Murali Karnam, Balázs Faludi, Nicolas Gerig, Georg Rauter, and Philippe C Cattin. Patient positioning by visualising surgical robot rotational workspace in augmented reality. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, pages 1–7, 2021.

[146] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[147] Jing Zhang, C-H Yan, C-K Chui, and S-H Ong. Fast segmentation of bone in ct images using 3d adaptive thresholding. *Computers in biology and medicine*, 40(2):231–236, 2010.

[148] Xiaomeng Zhang, Jing Wang, and Lei Xing. Metal artifact reduction in x-ray computed tomography (ct) by constrained optimization. *Medical physics*, 38(2):701–711, 2011.