

Reliability of Uncertainty for Assessing the Generalisability of Deep Learning Segmentation for Head and Neck Organs at Risk

Joëlle E. van Aalst

A Master's thesis

Submitted to the Faculty of Science and Technology
For the degree of Master of Science

in

Technical Medicine

Medical Imaging and Interventions

**UNIVERSITY
OF TWENTE.**

University of Twente,
The Netherlands



umcg

University Medical Centre Groningen,
Department of Radiation Oncology,
The Netherlands

September, 2024

Graduation Committee

Chairman

Prof. dr. Christoph Brune
Mathematics of Imaging & AI (MIA)
Department of Applied Mathematics, University of Twente

Technological Supervisor

Dr. Jelmer M. Wolterink
Mathematics of Imaging & AI (MIA)
Department of Applied Mathematics, University of Twente

Technical Medicine Supervisor

Dr. Sanne V. van Dijk
Department of Radiation Oncology
University Medical Centre Groningen

Daily Supervisor

Dr. Charlotte L. Brouwer
Department of Radiation Oncology
University Medical Centre Groningen

Medical Supervisor

Dr. Roel J.H.M. Steenbakkers
Department of Radiation Oncology
University Medical Centre Groningen

Process Supervisor

MSc. Elyse M. Walter
Faculty of Science and Technology (TNW)
Technical Medicine
University of Twente

Extra Supervisor

Dr. Tomas Janssen
Department of Radiation Oncology
Netherlands Cancer Institute

External Supervisor

Dr. Jasper J. Homminga
Faculty of Science and Technology (TNW)
University of Twente

Preface

Thank you for the sun, the one that shines
on everyone who feels love

Oasis

Through it all, I was happy, and for that, I have to thank my supervisors for their support, which has been a guiding light throughout this thesis. **Jelmer**, your deep technical insights and quick responses have been invaluable in raising the quality of this thesis. **Charlotte**, our weekly meetings provided me with a waymark to keep me on the right track in an area where I was easily distracted by the wide range of research possibilities. **Tomas**, your vision helped to guide the research to solid conclusions from the data and find a useful place in the field of radiotherapy.

Roel, your unconditional trust in giving me responsibility for patient care every week and your patience and kindness allowed me to fulfil the desire to care for patients alongside technical research. **Christoph**, thank you for your insightful comments which often expanded my local view on the research to the bigger picture of previous research and other domains. **Sanne**, I really enjoyed our time together in Lyon and learned a lot from your practical experience in radiotherapy research and your role as a technical physician. **Elyse**, thank you for your personal guidance, in times when I was unsure or lost, I could count on you for reassurance.

Lastly, a heartfelt thank you to my **colleagues**, **housemates**, **friends** and **family**, in this thesis, and also in myself, I truly see a mosaic of everyone who supported me.

Abstract

Deep learning (DL) segmentation models for organs at risk (OARs) in radiotherapy have reached the accuracy of manual segmentation. The high-risk environment of radiotherapy has led to cautious adoption of these autosegmentation models, affecting their optimal deployment. During manual quality assurance (QA) of DL segmentation, many corrections are made, especially within 2 mm, with fluctuating, but often minimal effect on dose planning. These corrections partially eliminate the efficiency gain proposed by DL segmentation models and re-introduce inter- and intra-observer variability. The use of uncertainty in automated or semi-manual quality assurance of autosegmentation results is an emerging field. However, there is no consensus on the optimal method and metrics to compute uncertainty maps. Furthermore, there is disagreement on the theoretical understanding of uncertainty representation in radiotherapy and disentanglement of uncertainty for out-of-distribution detection and data uncertainty visualisation. For out-of-distribution detection, no practical research with real-world radiotherapy data has been performed in radiotherapy.

This thesis evaluates the reliability of common uncertainty estimation methods and metrics. Additionally, we test their ability to distinguish between in-distribution and out-of-distribution data by withholding patient subsets during training, defined by radiotherapy-specific constraints (metal artefacts, a bolus and a bitelock). Using an nnU-Net segmentation model for 20 head and neck cancer organs at risk, we computed uncertainty estimations with Monte Carlo dropout, deep ensemble modelling, and test-time-augmentation. We find that these methods produce trustworthy uncertainty maps based on reliability diagrams and model performance stability. However, the quality between uncertainty metrics varies greatly, with entropy outperforming mutual information and variance. Finally, we show that uncertainty can effectively detect out-of-distribution cases, particularly when the visibility or shape of organs-at-risk is influenced between subsets.

Keywords: uncertainty, segmentation, out-of-distribution detection, generalisability of AI, head and neck cancer, radiotherapy, Monte Carlo dropout, deep ensemble modelling, test-time augmentation

List of abbreviations

AI	artificial intelligence
APL	added path length
CC	clinical contour
CNN	convolutional neural network
CT	computed tomography
DL	deep learning
DLC	deep learning contour
DSC	Dice-Sørensen coefficient
ECE	expected calibration error
FP	false positive
FN	false negative
HD	Hausdorff distance
HNC	head and neck cancer
HU	Hounsfield unit
ID	in-distribution
MC-dropout	Monte-Carlo dropout
ML	machine learning
NN	neural network
NTCP	normal tissue complication probability
OAR	organ at risk
OOD	out-of-distribution
PTV	planning target volume
QA	quality assurance
RT	radiation therapy
RTSTRUCT	radiotherapy structure set
RTT	radiotherapy technologist
TN	true negative
TP	true positive
U-E	uncertainty error overlap
UMCG	University Medical Centre Groningen
nnU-Net	no-new UNet

Contents

1	General Introduction	1
1.1	Contributions	3
1.2	Thesis outline	3
2	Clinical Background	4
2.1	Radiobiological effects	4
2.2	Head and neck anatomy and pathology	5
2.3	Segmentation for toxicity minimisation	8
3	Technological Background	9
3.1	Autosegmentation	9
3.1.1	Neural Networks	10
3.1.2	Convolutional Neural Networks	10
3.2	Safe AI implementation	11
3.3	Uncertainty in Deep Learning	13
3.3.1	Variational Bayesian approximation	13
3.3.2	Uncertainty computation	14
3.3.3	Uncertainty disentanglement	16
3.4	Previous work in uncertainty estimation for medical image segmentation . .	18
4	Head and Neck Cancer Organs at Risk Autosegmentation	20
4.1	Introduction	20
4.2	Method	21
4.2.1	Dataset and preprocessing	21
4.2.2	nnU-Net preprocessing	22
4.2.3	Model training	22
4.2.4	Geometric performance	22
4.2.5	Dosimetric performance	24
4.2.6	Performance evaluation	24
4.3	Results	25
4.4	Discussion	29
4.4.1	Model training	29
4.4.2	Geometric performance	29
4.4.3	Patient level geometric performance	30
4.4.4	Geometric performance comparison with existing autosegmentation models	30
4.4.5	Dosimetric performance	31
4.4.6	Limitations	31
4.5	Conclusion	32
5	Reliability Evaluation of Uncertainty Computation	33
5.1	Introduction	33
5.2	Background	34

5.2.1	Obtaining an output distribution: uncertainty methods	34
5.2.2	From samples to uncertainty map: uncertainty metrics	35
5.3	Methods	36
5.3.1	Theory	36
5.3.2	Experimental Setup	38
5.3.3	Autosegmentation performance and uncertainty map calibration . .	40
5.4	Results	40
5.4.1	Autosegmentation performance evaluation	40
5.4.2	Uncertainty map calibration evaluation	42
5.5	Discussion	45
5.6	Conclusion	48
6	Uncertainty for Out-of-Distribution Detection	49
6.1	Introduction	49
6.2	Background	50
6.3	Problem setup	51
6.4	Methods	52
6.4.1	Uncertainty estimation methods	52
6.4.2	Uncertainty estimation metric	52
6.4.3	Desiderata and experiments	53
6.5	Results	56
6.5.1	Analysis 1.1: Model performance on in-distribution (ID) patients . .	56
6.5.2	Analysis 1.2: Model performance on out-of-distribution (OOD) patients	56
6.5.3	Analysis 2.1: Uncertainty reliability	56
6.6	Discussion	61
6.6.1	Model performance	61
6.6.2	Uncertainty reliability	62
6.6.3	Uncertainty for OOD detection	62
6.6.4	Limitations	63
6.7	Conclusion	64
7	General conclusion and discussion	65
8	Future research	66
9	Appendix	75
9.1	Overview previous work in uncertainty computation for medical image segmentation	75
9.2	nnUNet network configuration and training parameter settings	77
9.3	nnUNet training commands	83
9.4	Patient level autosegmentation analysis	83
9.5	DIP vs DIE vs DIA	84
9.6	Extra geometric performance metrics nnUNet	85
9.7	Extra dosimetric performance metric nnUNet	86
9.8	nnUNet with and without mirroring	88
9.9	Wilcoxon signed rank test for geometric and dosimetric differences	89
9.10	Error localisation	91
9.10.1	Utilising uncertainty to locate errors	91
9.10.2	Error Localisation	91
9.11	OOD analysis: uncertainty map metrics	94

Chapter 1

General Introduction

Head and neck cancer (HNC) comprises a wide range of tumours in a complex anatomical and physiological environment ranging from the nasal cavity to the larynx. [1] The treatment for HNCs typically consists of a multimodal approach that may include surgical removal, radiation therapy (RT), chemotherapy, or a combination thereof.[2] RT often plays a critical role in HNC treatment.[3] To maximise target control and minimise RT-related toxicity, an optimal treatment plan based on a clinically accurate tumour- and organ at risk (OAR) segmentation is essential. However, the process of RT treatment planning requires a significant portion of physician and staff time. [1]

A fully automated workflow, including automated segmentation and automated treatment planning, shows promising potential as a method to reduce inter- and intra-observer variability and optimise time efficiency. deep learning (DL)-based segmentation (autosegmentation) has gained widespread attention, where many research groups and commercial parties, have published autosegmentation models, achieving high segmentation accuracy. [4, 5, 6] Thus shifting to the implementation-phase of OAR autosegmentation in RT.

However, optimal deployment of these models in the clinical workflow has been limited, possibly caused by the high-risk environment of RT. [7] In an RT workflow, faulty autosegmentation can introduce fatal errors in terms of incomplete target control or radiation-related toxicities. Segmentation quality directly translates to RT treatment plan quality and plan evaluation quality.[8] However, even when being incorrect, DL models present their output no other than when being correct. [9] Therefore, (manual) quality assurance (QA) of deployed DL model output at critical points in a fully automated radiotherapy workflow is still necessary for safe RT treatment delivery, see Figure 1.1.

Although autosegmentation has reached the accuracy of manual segmentation, [10], there is an existing reluctance to adopt DL segmentation as is during *manual* QA. This results in contour adjustments to the deep learning contour (DLC)s. A study by Brouwer et al. [11] has shown that oftentimes DLCs are corrected during manual QA, especially within 2mm. This partially eliminates the efficiency gain proposed by DL autosegmentation models [5] and re-introduces inter-observer and intra-observer variation. Furthermore, a prospect in RT is online adaptive treatment, where high-paced composition and evaluation of a treatment plan is necessary for real-time modifications to a patient's treatment plan, making autosegmentation and efficient (semi-)automated quality control inevitable. To address the obstacles with manual QA, *automated* QA has been approached. Automatically evaluating the output of a model can serve to consistently quantify the performance and suitability of model employment per patient (i.e. input).

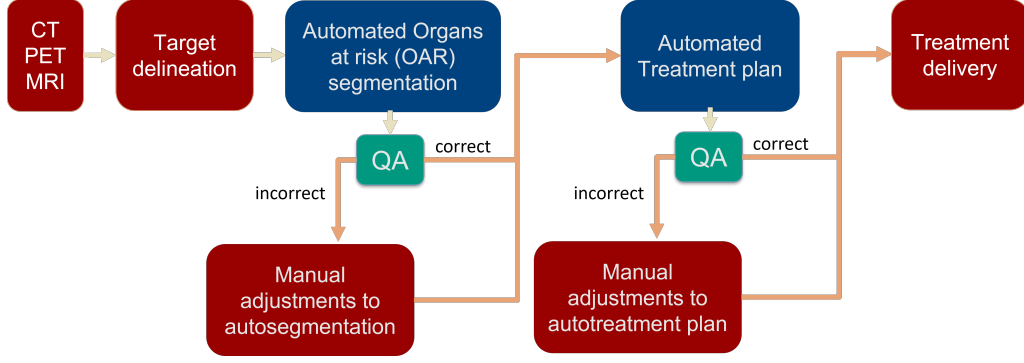


Figure 1.1: A fully automated radiotherapy workflow, including automated segmentation and treatment planning with QA steps ensure high-quality treatment delivery

Contemporary methods for automated QA generally include models that have been developed to identify an incorrect segmentation based on texture features [12, 13] or shape features [14, 15, 16, 17, 18] or a combination of both [19, 20, 21]. The probability of a DLC being correct, is based on comparing feature values with the feature probability distributions of previously approved contours. A combination of features is usually explored with an machine learning (ML) model. DL can also be used for automated QA, by validating an autosegmentation with a second segmentation algorithm, [22, 23, 24, 25, 24] a reverse segmentation algorithm [26, 27] or an algorithm specifically trained for QA [28, 29].

However, essentially what these methods do, is determine the possibility of an autosegmentation (i.e. model output, $\hat{y}$) being correct by comparing the features of the appearance to a set of previously accepted segmentations (i.e. validated model outputs, Y). In other words, they assess the *representativeness* of a model output ($p(\hat{y}|Y)$). A more mathematically grounded tool to reason about autosegmentation output quality is *uncertainty*, which assigns an individual confidence level to the stated model output.

Applying Bayesian probability theory to neural network (NN)s allows to effectively capture and represent uncertainty of a model output. [30, 31] The key idea is to find the probability of the predicted model output, $\hat{y}$, for an input, x , through the posterior distribution of the model parameters, w , given the training input and output set, $D = [X, Y]$: $p(w|D)$. The posterior distribution is used to obtain the probability of a new model output by averaging over all possible predictor models in the hypothesis space, W : $\int_w (p(\hat{y}|x, w)p(w|D))$. [32] Bayesian neural networks, as proposed by MacKay [30] and Neal [31], obtain the posterior distribution of a model by placing distributions over (a finite number of) model weights. This approaches a Gaussian process.

Applying uncertainty computation to DL models in medical computer vision is an emerging field. [33]. Various Bayesian approximation methods, such as Monte-Carlo dropout (MC-dropout) [34], Deep Ensemble modelling [35] and Test-time augmentation [36, 37], have been increasingly employed. However, no consensus has been reached on a single method and metric to compute uncertainty on a segmentation model. [33] Furthermore, a concept of disputation is the translation of the theoretical uncertainty behaviour to a practical level, especially the disentanglement of uncertainty into model uncertainty and data uncertainty. [38] Van den Berg et al. [39] and Barragan-Montero et al. [7] state that *data uncertainty* represents the stochastic component in the true labels where Meijerink et al. [40] and Stahl et al. [41] apply data uncertainty for OOD detection of a model input. While OOD detection is defined by van den Berg et al. [39] and Barragan-Montero et al. [7] as *model uncertainty* and not *data uncertainty*. According to Leousi et al. [42], *model uncertainty* should be applied to show where regions have been wrongly segmented.

Given the wide range of uncertainty computation techniques and the gap between theoretical understanding of uncertainty computations and real-world application, the use of uncertainty in automated QA for RT remains limited. This thesis evaluates the reliability of the most commonly used uncertainty methods (Deep Ensemble modelling, MC-dropout and Test-time augmentation) and metrics (entropy, mutual information and variance) for an RT application, and explores how uncertainty can assess the generalisability of a DL segmentation model in a real-world use-case of OOD detection. By enhancing the understanding of uncertainty in a clinically relevant way, this research aims to support the practical implementation of uncertainty measures in automated QA. Ultimately, this could enhance the efficiency and reliability of treatment planning and evaluation. Thereby improving patient care and reducing radiation-related toxicity.

The objective of this research is to explore the following research question: *How can uncertainty be reliably applied to analyse the generalisability of a deep learning autosegmentation model for head and neck cancer organs at risk in radiotherapy?* With subquestions:

1. How reliable are the most commonly used uncertainty computation methods and metrics for head and neck Cancer organs at risk autosegmentation?
2. Can uncertainty be applied to capture out-of-distribution data that an autosegmentation model is not trained on?

1.1 Contributions

The contribution of this thesis is threefold. In the first part, we train a traditional convolutional neural network (CNN)-based network, no-new UNet (nnU-Net), [43] for HNC OAR autosegmentation and compare its performance against state-of-the-art results found in literature. In the second part, we implement the three most popular uncertainty computation methods (MC-dropout [34], Deep Ensemble modelling [35] and Test-time augmentation [36]) and metrics (entropy, mutual-information and variance). We evaluate the trustworthiness of these uncertainty methods and metrics based on stable model accuracy performance and reliability diagrams. In the third part, we explore the translation between theoretical and practical understanding of uncertainty for OOD detection in RT with a set of data experiments. By influencing the data availability during model training (i.e. excluding specific subsets of patients), behavioural changes in model performance and uncertainty computation are expected. The resulting uncertainty maps and autosegmentation results provide a practical understanding of uncertainty computations. This enables investigation of the usability of uncertainty computations in automated QA.

1.2 Thesis outline

First, Chapter 2 provides a background on the clinical setting of this study: radiotherapy for HNC. Using this information, the importance and directives of QA in RT can be understood and applied to uncertainty computation evaluation. In Chapter 3 the theoretical technological background of autosegmentation and uncertainty along with its application in various Bayesian approximation methods is explored. In Chapter 4, firstly the choices and methodology for the autosegmentation network will be explained, after which the results are shown and discussed. In Chapter 5 the same will be done for the different uncertainty computations methods and metrics along with an analysis of their reliability. In Chapter 6 the data experiments to evaluate the practical understanding of uncertainty for OOD detection are presented in a similar manner. Finally Chapter 7 provides the conclusion of our research and Chapter 8 presents ideas for future research.

Chapter 2

Clinical Background

The principle of (curative) RT for the treatment of cancer is to maximise tumour control and minimise radiation toxicity in healthy tissue. For this, ionising radiation is used. [44] Ionising radiation can be found in the form of uncharged particles (e.g. photons) and charged particles (e.g. protons). Both types of radiation have the property to deposit energy in the medium it traverses through until it is stopped by absorption.

2.1 Radiobiological effects

The absorption of ionising radiation in biological tissue is not without consequences. The interaction causes ionisations and excitations in the atoms of biological material. In (organic) molecules in a cell, small structure changes can occur when ionisations or excitations occur. However, the biological effect of radiation is predominantly caused by damage to DNA molecules. The most prominent molecule in a cell is water. Ionising radiation causes the production of free (hydroxyl) radicals in a cell by breaking apart water molecules into ion pairs (radiolysis). When these free radicals hit vital molecules of a cell, for example, DNA molecules, a break in the DNA strand is caused. [45, 44]

A cell has various mechanisms to repair DNA damage. If the DNA is fully repaired, the effect of radiation on the cell remains absent. However, if the repair is incomplete, incorrect or too slow, a cell cannot reproduce anymore and the cell is traversed into programmed cell death (apoptosis). Not all phases of a cell are evenly sensitive to radiation damage. A cell is most sensitive when it is in the mitosis phase. This makes cells with a high cell division rate and undifferentiated cells more susceptible to radiation damage (i.e. cancer cells). [45]

Apoptosis of cancer cells is desirable in RT, however cell death of healthy tissue is not. To pursue this, limiting the total radiation dose on OARs is essential. Healthy organs have a specific radiation tolerance, above which damage can occur (for parallel organs) or will occur (for linear organs). Affecting the patient's quality of life during and after treatment. Fractionation of radiation treatment also reduces the damage to healthy tissue, as then the tissue is given time to repair damage and repopulate the damaged tissue with new cells. The process of cell repair takes around six hours. So by irradiating a patient daily with a fractionated dose, repair can occur. For HNC patients, an RT treatment plan typically consists of 33-35 fractions of 2 Gy. [45, 44]

2.2 Head and neck anatomy and pathology

The head and neck area is a complex anatomical region, consisting of many organs susceptible to radiation damage. In a RT treatment plan, the regions of interest encompassing healthy tissue are collectively joined in a list of OARs. Consensus for 25 head-and-neck OARs has been established by an international group of radiation oncologists [46]. See Table 2.1 for an anatomical and functional description of these OARs and Figure 2.1.

Excessive radiation damage to the OARs mentioned in Table 2.1 can lead to a variety of radiation induced toxicities/complications that can be linked to loss of their function. Effects can be both acute and long-term. This significantly impacts patient's quality of life during and after treatment. [47] The most common side effects of RT for HNC patients are dermatitis, xerostomia (i.e. dry mouth), sticky saliva, dysphagia (i.e. difficulty swallowing), mucositis and hypothyroidism. However, other, less common, symptoms can arise when dose distribution to specific OARs is suboptimal, such as eye damage and nervous system dysfunction can also arise. These complications will now be addressed in more detail on a pathological level.

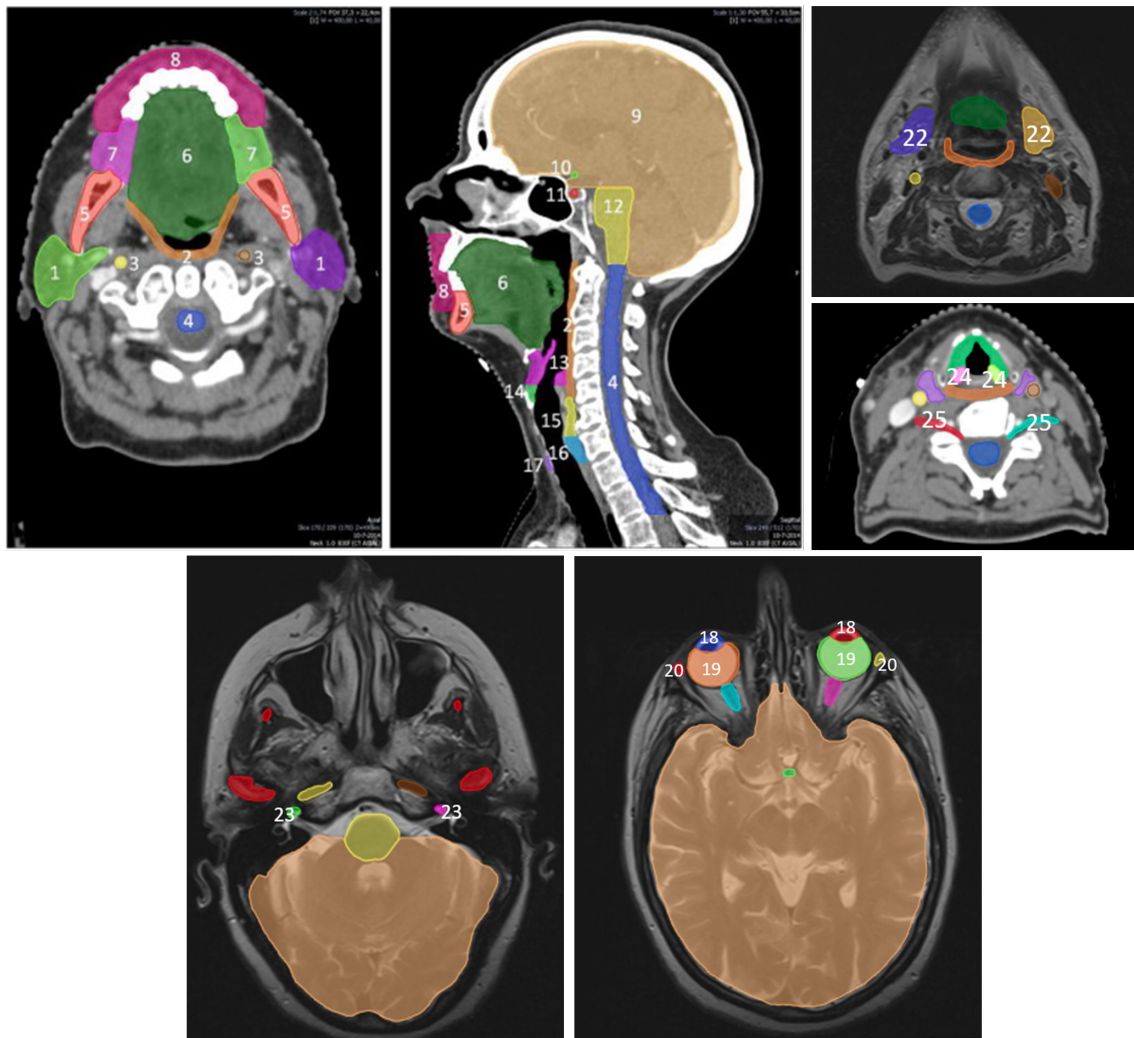


Figure 2.1: HNC OAR visualisation on a CT-scan. [46]

Table 2.1: Head and Neck Cancer Organs at Risk

Number in Figure 2.1	Organ at risk	Function/description
1	Parotid glands	Largest salivary gland.
2	Pharyngeal constrictor muscles	Constrict the pharynx to pass food through the oesophagus during swallowing
3	Carotid arteries	Arteries supplying the brain with blood
4	Spinal cord	Part of the nervous system that connects the brain to the peripheral nervous system
5	Mandible	Lower jaw bone, supporting the lower teeth
6	Extended oral cavity	Mouth
7	Buccal mucosa	Inner cheek lining, mucous membrane of mouth
8	Lips	Surround the oral fissure functioning as valves for what enters and exits the mouth and respiratory tracts
9	Brain	Centre of the nervous system which performs higher functions
10	Optic chiasm	Location where the optic nerves cross
11	Pituitary gland	Endocrine gland which regulates growth, metabolism and reproduction
12	Brainstem	Connection between the cerebrum and spinal cord
13	Supraglottic larynx	Part of the larynx, voice box, above the glottis (the glottis during swallowing of food to close the trachea and force the content to the oesophagus)
14	Glottic area	Middle part of the larynx, where the vocal cords are situated
15	Cricopharyngeal inlet	Opening of the cervical oesophagus
16	Cervical oesophagus	Most superior part of the oesophagus which is used to transport food from the pharynx to the stomach
17	Thyroid gland	Endocrine gland which produces thyroid hormone (controls metabolism rate, growth, temperature control and heart and muscle function) and calcitonin (controls calcium metabolism)
18 & 19	Eyeball (anterior and posterior segment respectively)	Contains optical apparatus of visual system
20	Lacrimal gland	Gland which secretes lacrimal fluid which protects the cornea and eyeballs from injury and irritation
21	Optic nerve	Nerve regulating sense of vision
22	Submandibular glands	Salivary glands below the mandible bone on both sides of the jaw
23	Cochlea	Shell-shaped part of the bony labyrinth in the inner ear containing the cochlear duct
24	Arytenoids	Cartilage surrounding the larynx which assist the vocal cords
25	Brachial plexus	Network of nerves which innervate the upper limbs

Dermatitis

During external radiation treatment, it is inevitable that the skin receives a part of the deposited dose. Radiation reacts with the epidermal layer of the skin, but can also damage deeper layers. Leading to a range of possible skin reactions from acute radiation dermatitis to permanent loss of elasticity/fibrosis, depending on the fractionation schedule and total dose to the skin. The patient will recognise their skin getting red and dry. With increasing deposited dose, blisters, desquamation or hair loss can also occur, making the patient susceptible for skin infections. [45, 48] Apart from hair loss, symptoms for this RT complication, are mainly an acute toxicity, that heals relatively quickly.

Xerostomia/Sticky saliva

RT can irreversibly damage salivary gland cells in the parotid and submandibular glands. This can lead to less saliva or thick and sticky saliva being produced. These are the most common acute and long-term side effects of RT in HNC. Minimising the absorbed dose in the parotid and submandibular glands can lead to a clinically significant reduction in the incidence and severity of xerostomia and/or thick saliva. [49, 50]

Dysphagia

For swallowing, a complex corporation of a large set of muscles and structures in the oral cavity, pharynx, larynx and oesophagus is necessary. This includes the supraglottic larynx and glottic area, the submandibular glands, oesophagus, mandible and among other muscles, the pharyngeal constrictor muscles. [50, 51] Oedema (acute toxicity) or fibrosis (long-term toxicity) can impair swallowing function, but it can also be caused by neuropathy. Minimising dose distribution in these OARs has a clinical benefit for the patient, by decreasing the chance of this RT-related toxicity. [52]

Damage to these tissues can also lead to a variety of other symptoms including dysarthria (difficulty speaking), dysphonia (abnormal voice) trismus (difficulty opening the mouth)). [53]

Mucositis

Similarly to the outer lining (i.e. the skin), the inner lining of the human body (i.e. the mucosa) experiences radiation damage as well. During RT treatment, a patient can experience pain and a burning sensation in the mouth and throat that is caused by radiation damage to the inner lining (i.e. mucositis) of the oral cavity, supraglottic larynx and glottis. This sensation can be aggravated by a dry mouth caused by radiation damage to the salivary glands. Mucositis is mainly associated with radiation dose distribution to the oral tissues (oral cavity and tongue). [54] This is an acute radiation therapy complication that resolves once the mucosa heals.

Hypothyroidism

Radiation damage to the thyroid gland can cause complications related to metabolic processes, as the thyroid and pituitary gland function can be impaired by dose distribution. Radiation dose to the thyroid is directly related to hypothyroidism incidence. Thyroid and pituitary gland dysfunction cause a higher concentration of thyroid stimulating hormone (TSH) and a lower concentration of free thyroxine (FT4) leading to hypothyroidism. This can lead to a variety of symptoms including fatigue, drowsiness, cold intolerance, weight gain, constipation and also cardiovascular diseases in severe cases. [55] This is a late occurring RT toxicity.

Eye damage

Radiation damage to the optic chiasm, optical nerves, eyeball and lacrimal gland can cause a variety of vision-related complications such as cataracts, retinopathy or overall vision loss can impede as a radiation complication because of radiation damage to structures such as

the optic chiasm, optical nerves, eyeball and/or lacrimal glands. [49] These are long-term, deterministic RT-related toxicities. Thus, strict dose thresholds must be applied to the related OARs to prevent occurrence. [45]

Nervous system dysfunction

A variety of structures from the central and peripheral nervous system are located in the head and neck area, including the cerebrum, brainstem, spinal cord and brachial plexus. Neurological damage such as pain, memory loss and altered brain function can be caused by radiation damage to the nervous system or oedema of surrounding tissue. Radiation damage to the brachial plexus, including fibrosis, can cause paraesthesia (abnormal sensation in extremities), dysesthesia (disordered touch sensation) and movement loss. [49] Similarly to eye damage, this is also a deterministic RT toxicity that can be prevented with adequate dose thresholds to involved OARs. [45]

2.3 Segmentation for toxicity minimisation

To prevent and limit the possibility of radiation toxicity, dose to OARs is minimised by a list of dose/volume objectives and constraints during RT treatment planning. To make and evaluate a treatment plan, these OARs are delineated. For the safety of radiotherapy, reliable and accurate delineation of the tumour *and* OARs is thus of utmost importance. However, manual autosegmentation has been shown to be time-consuming and inconsistent in terms of inter- and intra-observer variability. Thus leading to the introduction of automated segmentation.

Chapter 3

Technological Background

Artificial intelligence (AI) has proven its effectiveness in automating repetitive, labour-intensive tasks and analysing large datasets for patterns, with medicine being a prominent application. Particularly employment of AI in the interpretation of medical images has proven its value, by showing high accuracy and having the potential to decrease clinical workload. [56, 57] A key application is semantic segmentation, a computer vision problem where, in the case of 3D medical imaging, each voxel within a volume is classified according to the object the voxel represents. This voxel-wise classification enables the automatic delineation of structures on a 3D medical image. In radiotherapy, autosegmentation is commonly used to refer to DL segmentation. In this thesis 'autosegmentation' and 'DL segmentation' will be used interchangeably.

3.1 Autosegmentation

The initial breakthrough in autosegmentation for medical applications occurred with the development of atlas and model-based methods. These approaches register a new patient image to a set of reference images with a predefined ground-truth segmentation. Through registration, transformation and refinement, an automated annotation is obtained. [58].

While at the 2015 MICCAI challenge for head and neck OARs segmentation, only atlas and statistical model-based approaches had been submitted, the focus of segmentation in RT has now shifted towards DL approaches. These feature-based learning techniques have achieved higher accuracy and can accommodate a broader range of patients, as medical data with artefacts and patients with diverse anatomy can be included. The most popular DL methods in medical imaging have till now been CNNs. [58, 57, 4, 5] Recently, however, transformers - originally developed for the language learning domain [59] - have been integrated into the vision domain as well. [60] The added advantage of transformers is that features of long-range relationships can be explored through attention-based mechanisms instead of the locally operating CNNs.

In this section, a background on the mechanisms and existing applications of CNNs will be provided. Although it is of utmost interest to compare the functionality between conventional CNN networks and transformer networks applied on HNC OARs, there was simply not enough time to focus on both networks in this thesis. However, this is a very interesting area for future research. Before expanding on the functionality of CNNs for autosegmentation, firstly a general background on NNs will be provided.

3.1.1 Neural Networks

NNs are mathematical models inspired by biological neural networks, designed to represent complex relationships and processes. NNs consist of interconnected units, neurons. Each neuron can be seen as a parameterised unit that receives multiple inputs, applies learned weights to them and produces an output based on a threshold value known as the bias.

The goal of training an NN is to find the optimal weights and biases that maximise the network's performance on a given task. This is achieved by comparing the network's predictions to the true target value with a loss function. The loss function quantifies the error between the predicted and actual values. The gradient of the loss function with respect to the network's parameters is computed, to determine how changes in these parameters will affect the loss. This gradient information guides the adjustments to the weights and biases needed to minimise the error and improve the network's predictions. This process is called backpropagation and is performed through an optimisation algorithm.

3.1.2 Convolutional Neural Networks

In a medical imaging application, structures are represented by sets of voxels. Analysing each voxel individually is inefficient and can lead to the loss of contextual information. CNNs address this by employing convolution operations. A kernel (i.e. convolution operation) is applied to input data to produce a feature map. Pooling operations are then employed to aggregate feature maps, these use permutation invariant functions that further reduce feature map size by computing combined features. This is the encoder part of the network. Pooling operations simultaneously decrease the number of parameters that need to be analysed in the model and ensure translation invariance [61, 62].

U-Net

A popular implementation of CNNs for segmentation tasks is the U-Net architecture. [63] A U-Net is designed with an encoder and a decoder to perform image-to-image tasks. To recover the reduced size of the feature maps, the decoder performs up-sampling. This allows the network to combine low-level details with high-level context and ensures that the model does not have to be trained for each input patch separately. [63] Combining the encoder and decoder leads to a U-shaped network that has the same shaped pixel-level input and output that can analyse the data on various magnitudes of context.

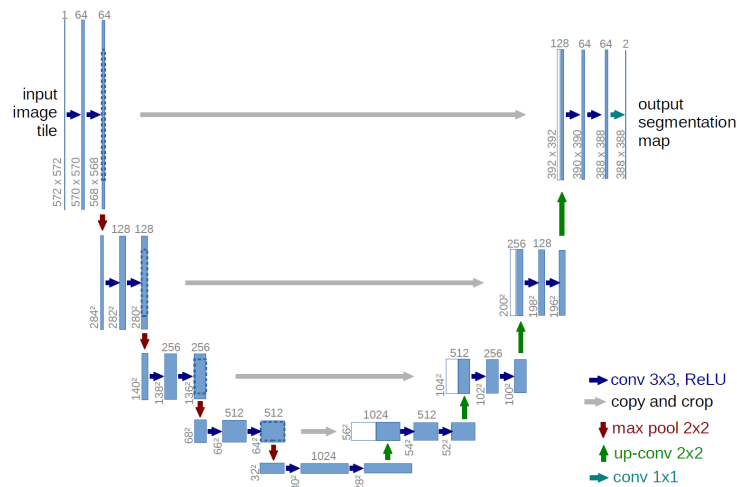


Figure 3.1: U-Net architecture, adapted from [63]

3.2 Safe AI implementation

All models are wrong, but some are useful

George Box

A fully automated workflow, including automated segmentation and automated treatment planning, shows promising potential as a method to reduce inter- and intra-observer variability and optimise time-efficiency. An increasing number of deep learning (DL) methods aimed at automating organ at risk (OAR) segmentation and treatment planning, is being integrated in routine clinical practice for RT due to its high-performing accuracy. [64] However, it is important to remember that NNs remain approximations of complex real-world processes, such as segmentation. As highlighted by the quote at the beginning of this section, all models are merely a mathematical representation that approximate the most optimal parameters that transform input to an output, obtained using a limited set of training data.

A consequence is, that, even when being incorrect, DL models present their output as if it were the truth. In an RT workflow, faulty autosegmentation can introduce fatal errors in terms of incomplete target control or radiation-related toxicities. [9] Segmentation quality directly translates to radiation therapy treatment plan quality and treatment plan evaluation quality.[8] Therefore QA at critical points in a fully automated radiotherapy workflow is essential. A model is only useful when you also know when it is wrong.

QA is, at the moment, often performed manually or semi-automatically. However, this re-introduces inter- and intra-observer variability and diminishes the efficiency gain. A study by Brouwer et al. [11] for example has shown that with manual autosegmentation QA, adjustments to autosegmentations are still often performed, especially within 2 mm. Automatically quantifying the uncertainty of autosegmentation could assist in a more efficient automated RT workflow for autosegmentation evaluation. While simultaneously preserving autosegmentation quality consistency.

The purpose of automated QA in a fully automated RT workflow is to automatically perform quality prediction of autosegmentation contours. Automated QA for autosegmentation in radiotherapy has been extensively explored in published research, see Table 9.2. Approaches can generally be divided into ML- and DL-based methods. Where with ML the focus has been mainly on image features (also called texture features)[12, 13] and shape features [14, 15, 16, 17, 18] or a combination thereof [19, 20, 21], with DL the main focus has been on autosegmentation validation with a second segmentation algorithm [22, 23, 24, 65, 25, 24], a reverse segmentation algorithm [26, 27] or a DL algorithm specifically trained for QA [28, 29]. For an overview see Table 9.2

What these automated QA methods however do, is determine the possibility of an output being correct by comparing the features of the appearance to a set of previously validated segmentations. In other words, they try to assess the representativeness of the output. However, autosegmentation (i.e. output) quality does not solely depend on representativeness to a previous set of (clinically approved) segmentations (i.e. prior outputs). There is also a degree of independence in the quality of the autosegmentation resulting from the individual patient (i.e. input) and uncertainty in the model itself.

Author	Year	AI	Structures	Approach	Evaluation
Zhang et al. [12]	2019	ML	Pancreas head and duodenum	Texture features from subregions in a decision tree model.	Sens. 0.85-0.92 Spec. 0.91-0.92
Luan et al. [13]	2023	ML	18 HNC OARs	Shell-based (an)isotropic radiomic texture features in 12 ensemble machine learning models.	Acc.: 0.83-0.97
Chen et al. [14]	2015	ML	8 HNC OARs	Geometric features and spatial relationship of neighbouring structures in a geometric attribute distribution model inspired by PCA.	Sens.: 0.88 Spec.: 0.83
Altman et al. [15]	2015	ML	9 HNC OARs	Geometric features in a knowledge database.	Sens.: 0.95 Spec.: 0.81
Hui et al. [16]	2018	ML	7 lung OARs	Volumetric features (computed in 3D) in a knowledge database based on correct autosegmentations	Sens.: 0.68-0.87
Duan et al. [17]	2023	ML	4 HNC OARs	Geometric features in 4 ensemble machine learning models.	Rec.: 0.73-0.84 Prec.: 0.76-0.89
Wooten et al. [18]	2022	ML	Kidneys	Geometric features in a random forest model.	Sens.: 0.91 FPR: 0.11
McIntosh et al. [19]	2013	ML	9 Whole body OARs and targets	Geometric and texture features in a conditional random field on top of a random forest.	Sens. 0.85-0.92 Spec. 0.91-0.92
Papaconstadopoulos et al. [20]	2021	ML	Left and right parotid	Geometric and texture features in an isolation forest model.	Sens.: 0.93-0.99 Spec.: 0.73-0.79
Chen et al. [28]	2020	DL	Breast cancer CTV	A ResNet-101 QA model with the CT, segmentation probability (autosegmentation CNN output) and uncertainty maps as input. DSC is the output.	F1: 0.81-0.98 AUC: 0.88-0.96
Su et al. [29]	2022	DL	Bladder	A 3D-CNN/MV-CNN QA model with multiple views of the 3D mesh (binary pattern operator, canny operator) as input and quality prediction as the output.	Sens.: 0.91 Spec.: 0.95
Rhee et al. [22]	2019	DL	16 HNC OARs	A secondary DL autosegmentation model (V-Net/FCN-8s combination) evaluates the primary segmentation with DSC.	Sens.: 0.08-0.92 Spec.: 0.67-1.00
Men et al. [23]	2020	DL	4 lung OARs	A secondary DL autosegmentation model (CNN) evaluates the primary segmentation with DSC and HD.	Sens.: 0.95-1.0 Spec.: 0.94-0.98
Min et al. [24]	2021	DL	Prostate CTV and OARs	A secondary DL autosegmentation model (V-Net/FCN-8s combination) evaluates the primary segmentation with DSC, HD MSD and surface-DSC (1-10 mm) with a SVM.	Sens.: 0.73-0.93 Spec.: 0.86-0.95
Claessens et al. [25]	2022	DL	Prostate CTV and OARs	A secondary DL autosegmentation model (DLCExpert [Mirada Medical Ltd., Oxford, United Kingdom]) evaluates the primary segmentation with surface-DSC and APL	Sens.: 0.5-1.0 Prec.: 0.64-1.0
Valindria et al. [26]	2017	DL	15 whole body OARs	Reverse classification accuracy using a single-Atlas, Atlas Forests and CNNs with DSC	Acc.: 0.88-0.93
Robinson et al. [27]	2018	DL	Cardiac OARs	Reverse classification accuracy using a 3D ResNet-50 with DSC	Acc. 0.91
Nourzadeh et al. [21]	2022	DL	17 whole body OARs	Geometric, spatial and intensity features in classifiers to an auto-encoder.	Spec.: 0.75

Table 3.1: Summary of previous work in autosegmentation QA using ML and DL methods. AI = Artificial Intelligence, ML = Machine Learning, DL = Deep Learning. CTV = Clinical Target Volume, CNN = Convolutional Neural Network. MV = Multi-view. DSC = Sørensen-Dice similarity score. HD = Hausdorff distance. MSD = mean surface distance. APL = added path length. Sens. = sensitivity. Spec. = specificity. Acc. = accuracy. Prec. = precision. Rec. = recall. FPR = false positive rate.

3.3 Uncertainty in Deep Learning

Applying Bayesian probability theory to a DL model (Bayesian Deep Learning) does provide the mathematically grounded tools to reason about model uncertainty. [34] Bayesian probability theory revolves around updating the probability of a hypothesis (output), y , as more information (evidence) becomes available from the data, D , and its relation to the output:

$$p(y|D) = \frac{p(D|y) \cdot p(y)}{p(D)} \quad (3.1)$$

Bayesian probability theory on NNs allows to effectively represent uncertainty in the model by introducing probability distributions over the model's weights. This idea is key in Bayesian inference, where it is the object to find the posterior distribution, of the model output, y , given the certainty of previous information. This information is seen as the certainty of the predictor model's weights, w , based on the training input and output set, $D=[X, Y]$: $p(w | D)$. [66] Bayes' theorem describes how to do this:

$$p(w|D) = \frac{p(w) \cdot p(D|w)}{p(D)} = \frac{p(D|w)p(w)}{\int_{W'} p(D|w')p(w')dw'} \quad (3.2)$$

The posterior distribution of the model weights, $p(w | D)$, represents the state of knowledge of the predictor model parameters, learned from the training data set, D . Obtaining this posterior distribution, enables probabilistic prediction of a distribution over the predicted output, $\hat{y}$, for a new data point, x , by averaging over all possible predictor models, W , in the hypothesis model space [34, 66, 39]:

$$p(\hat{y}|x, D) = \int_W p(y|x, w)p(w|D)dw \quad (3.3)$$

A Bayesian Deep Learning autosegmentation model not only performs semantic segmentation but also provides uncertainty estimates per voxel. This uncertainty is calculated individually for each model output, by assessing the confidence given all possible model weights that the model could have, based on the training input and output data.

3.3.1 Variational Bayesian approximation

However, computing the integral over all possible values of the model weights W , as is necessary per 3.2 to obtain $p(w|D)$, results in an immense computational cost and challenging inference. Gal et al. [34] have shown that *approximating* the true posterior with a numerical solution is possible with variational inference. With variational inference, the Kullback-Leibler divergence is minimised between a weight distribution parameterised by θ , $q_\theta(w)$, (i.e. the approximate distribution), and the true posterior of the model space given the training data, $p(w|D)$. This is equivalent to maximising the evidence lower bound score, which is intractable [34, 66, 67, 39]:

$$KL(q_\theta(w)||p(w|D)) \quad (3.4)$$

In other words, Bayesian approximation approximates the true posterior of the model's weights. Two popular methods for variational Bayesian approximation to obtain the distribution, are: Monte Carlo dropout (MC-dropout)[34] and Deep Ensemble modelling [35].

MC-dropout

Gal and Ghahramani’s research has played a crucial role in advancing methods for approximating Bayesian neural networks. [34] They stated that dropout at inference time is mathematically equivalent to an approximation of the probabilistic Deep Gaussian process through sampling-based inference. Dropout is a technique commonly used during training of a DL to avoid over-fitting. However, when applied during inference time, a collection of predictions can be sampled and averaged to compute model uncertainty due to the stochasticity of the forward passes. [34, 42] This method can be used to approximate the posterior distribution of the model’s weights.

Deep Ensemble networks

An alternative to MC-dropout for Bayesian approximation has been proposed by Lakshminarayanan et al. [35]. They demonstrated that a combination of predictions from an ensemble of multiple models, a Deep Ensemble, is capable of performing Bayesian inference. Deep Ensemble modelling can effectively represent the posterior distribution over a model’s parameters and quantify predictive uncertainty by using a proper scoring rule during training. The scoring rule evaluates the quality of the predictive distribution, $p(y|x, W)$ by comparing it to the true distribution, $q(y, x)$. Many common loss functions for NNs are already proper scoring rules, such as the Softmax cross entropy loss or the Brier score.[35]

Similarly to MC-dropout, Deep Ensemble modelling is a technique commonly used in *DL* model development to improve predictive accuracy. [68] However, during inference, Deep Ensembles have the added advantage that they use a more robust approach compared to MC-dropout by using regularised, fully developed *DL* models rather than ‘models’ originating from random node drops. [69]

3.3.2 Uncertainty computation

Deep Ensemble modelling and MC-dropout can thus conceptually both be seen as a way of sampling a set of model parameters that together approach the model hypothesis space distribution through their stochasticity. [70] A deeper mathematical explanation on the computation of uncertainty will now be provided. As an example MC-dropout will be used, however as it conceptually follows the same approach as Deep Ensemble modelling, the method can be easily translated.

A brief summary of the concepts of Bayesian Deep approximation is that the model likelihood (i.e. epistemic uncertainty), $p(y|f^W(x))$ is obtained through Bayesian inference using the posterior over the model weights, $p(W|D)$, which captures the set of plausible model parameters given the data. With $f^W(x)$ the representation of computing the model output for an input x . By marginalising over the approximate weights, the posterior distribution results in the prediction uncertainty. This computationally expensive process can be approximated using Monte Carlo integration and results in the probability of the output $\hat{y}$ being a certain class c given the input x and previous training data D . The probability distribution can be computed in the following manner directly from the set of model outputs:

$$p(y = c|x, D) \approx \frac{1}{T} \sum_{t=1}^T \text{Softmax}(f^{\hat{W}_t}(x)) \quad (3.5)$$

With T sampled masked model weights $\hat{W} \sim q_{\theta}^*(W)$, where $q_{\theta}(W)$ is the Dropout weight distribution, with θ the distribution parameters to be optimised (i.e. the parameters that define the weight distribution in this case)

As the output of the Softmax activation layer contain logits, they need to be converted to a confidence value with an uncertainty metric. Unlike regression, where the variance between samples can be used to translate this probability distribution to uncertainty, classification requires metrics that can handle categorical outputs effectively. Gal et al. [34] proposed the Shannon entropy and mutual information as an appropriate measure for classification problems.

Shannon entropy

The probability distributions obtained from sampling, as described in 3.5, can be used to compute the Shannon entropy $H(p)$. Shannon entropy quantifies the flatness of the probability distribution over the output space. It reflects the confidence in the model's prediction for the assigned class: $p(y = c|x, D)$. A high entropy value indicates a high level of uncertainty. [71] Gal et al. [34] described this metric as capturing *predictive uncertainty* (which includes both uncertainty arising from the model and data). It can be interpreted as the degree of uniformity in the model's prediction, reflecting the distribution of possible outcomes given the model's parameters and the data. [72]

$$H(p) = - \sum_{c=1}^C p \log p \quad (3.6)$$

With c ranging over all classes and p the probability obtained through Monte Carlo sampling by averaging all Softmax outputs as described in Equation 3.5

Mutual information

Mutual information captures model uncertainty only, according to Gal et al. [34]. It represents the difference between the posterior density function, $p(\hat{y}|x, W)$, and the prediction density function, $p(\hat{y}|x, D)$, i.e. what the model adds about the knowledge: [71]

$$\hat{M}I[\hat{y} | x, D] = \hat{\mathbb{H}}[y|x, D] - \frac{1}{T} \sum_{t=1}^T \sum_{c=1}^C p(y = c|x, \hat{w}_t) \log p(y = c|x, \hat{w}_t) \quad (3.7)$$

With $\mathbb{H}$ the predictive entropy, i.e. the Shannon Entropy, C the number of classes, T the number of Monte Carlo samples, $\hat{w}_t$ model weights at the repsective Monte Carlo sample t and p obtained from Equation 3.5

This metric captures information about the consistency across the samples. This measure is confident if the model was trained on enough similar data. [71]

Variance

A third option to obtain an uncertainty map is to directly compute the variance of the Monte Carlo samples. The sample variance is computed as the variance of several samples taken of the network's output compared against their average:

$$var = \frac{1}{T} \sum_{t=1}^T (\hat{y}_{i,t} - \hat{y}_{i,avg})^2 \quad (3.8)$$

With $\hat{y}_{i,t}$ the sigmoid output at voxel I for Monte Carlo sample t , $\hat{y}_{i,avg}$ the average prediction obtained from T Monte Carlo samples, according to $\sum_{t=1}^T y_{i,t}$

This metric captures information about the consistency across the samples. This measure is confident if the model was trained on enough similar data. This supposedly reflects epistemic uncertainty. [71]

Interpretation of uncertainty metrics

Nair et al. [71] provided an insightful toy example for better understanding of the three uncertainty metrics mentioned above, see Table 3.2. Here it becomes apparent that when there is a discrepancy between the entropy and mutual information or variance, the model is certain that its prediction is uncertain. Variance and mutual information capture information about the consistency (i.e. model uncertainty) and entropy captures information about the flatness of the distribution (i.e. total uncertainty).

Table 3.2: Toy data examples with various uncertainty metrics

Example	Samples	Variance	Entropy	Mutual information
1	[1.0, 1.0, 1.0, 1.0, 1.0]	0	0	0
2	[0.5, 0.5, 0.5, 0.5, 0.5]	0	0.693	0
3	[1.0, 1.0, 1.0, 1.0, 0.0]	0.2	0.5	0.5
4	[0.5, 0.5, 0.5, 1.0, 1.0]	0.075	0.611	0.195
5	[0.6, 0.7, 0.8, 0.8, 0.7]	0.007	0.522	0.057

Samples indicate an array of 5 outputs obtained with an uncertainty method. For the uncertainty metrics 1.0 represents most confident and 0 least confident. Adapted from Nair et al. [71]

3.3.3 Uncertainty disentanglement

The uncertainty of a DL model can thus be obtained in a relatively straightforward implementation of existing ML practices using MC-dropout or Deep Ensemble modelling and an appropriate uncertainty metric to translate the value to a confidence level. While various types of uncertainty have been introduced when describing the uncertainty metrics, clarification on the different types of uncertainty in a DL model will now be provided.

Kendall and Gal [38] described epistemic and aleatoric uncertainty for computer vision as 'model parameter uncertainty' and 'noise inherent in the observations' respectively. Epistemic uncertainty reflects OOD examples that fall outside the training domain, i.e. a lack of information for accurate prediction. This model uncertainty can be explained away by diversifying the training data or by building a more advanced DL model. While aleatoric uncertainty reflects the inherent stochasticity of the training environment, and can generally not be explained away, only if the stochasticity is reduced by for example a reduction in sensor noise.

However, it is important to note that the boundary between aleatoric and epistemic uncertainty is not always clear, particularly when leaving the domain of highly controlled toy data experiments.[39, 73] Nevertheless, Charpentier et al. [74] argued that a division of uncertainty into aleatoric and epistemic uncertainty is key to achieve the *DL* model properties for a reliable deployment in real-world applications. Without separation, model generalisation evaluation and OOD detection are unfeasible. Paraphrased, OOD detection as described by Charpentier et al. [74], can be seen as limits of the capability of a model to process input by the distribution a model was trained on. It can be identified with epistemic uncertainty. While generalisation performance assessment can be seen as the semantic segmentation uncertainty and can be identified with aleatoric uncertainty.

Van den Berg et al. [39] and Barragan-Montero et al. [7] have similarly argued that both uncertainty types are necessary to improve *DL* model implementation for semantic segmentation by providing the ability to track and fix issues independently. Their notions on uncertainty disentanglement and what the epistemic and aleatoric uncertainty signify in

medical image segmentation, along with notions provided by other articles, are summarised in item 3.3

Table 3.3: Summary of practical definitions of uncertainty disentanglement in medical image segmentation for RT

Aleatoric uncertainty	Van den Berg et al. [39]: aleatoric uncertainty is a representation of the inter-observer variability which is caused by the stochastic component in labels Barragan-Montero et al. [7]: aleatoric uncertainty is caused by noise, inaccurate or low quality records and labels. Leousi [42]: aleatoric component of uncertainty is confronted by clinicians as well in terms of poor input image quality which hampers manual contouring, leading to variations in clinical segmentations. Meijerink et al. [40]: out-of-distribution input produces low aleatoric uncertainty Stahl et al. [41]: out-of-distribution input produces low aleatoric uncertainty	Can be improved by increasing quality of inputs (both training data and new unseen samples)
Epistemic uncertainty	Van den Berg et al. [39]: epistemic uncertainty arises from the fact that not the total structure of the data generating process is known, i.e., the ground truth model is not available and/or only a limited number of (noisy) observations are available for training Barragan-Montero et al. [7]: epistemic is a representation of when the training data does not represent the population under study well OR model does not interpolate/extrapolate well enough Leousi [42]: epistemic uncertainty shows which regions have been wrongly segmented or where challenging voxels occurred.	Can be improved by: 1. More data to better sample the problem 2. Using more appropriate architecture with improved learning abilities 3. Increasing the quality of the training set

Test-time augmentation

An option to compute the aleatoric uncertainty (i.e. uncertainty arising through the stochasticity in the data), is by using Test-time augmentation. Augmentation is again a technique already applied during training to increase the number of training examples and simultaneously introduce more variety into the data.[75] However, Hu et al. [76] state that the proposed method does theoretically not compute aleatoric uncertainty, but epistemic uncertainty.

Data augmentation includes operations such as flipping, scaling, rotating and cropping. This introduces an ensemble of output variations on which the aleatoric uncertainty related to image transformations and noise can be obtained. By determining the entropy of the predictions for a given image, the aleatoric uncertainty is obtained:

$$H(y_i|x) \approx - \sum_{c=1}^C \hat{p}_i \ln(\hat{p}_i) \quad (3.9)$$

With c ranging over all classes and $\hat{p}_i$ the probability of the i th voxel for class c as obtained through Test-time augmentation sampling by averaging all softmax outputs as described in Equation 3.5.

3.4 Previous work in uncertainty estimation for medical image segmentation

The above-mentioned techniques (MC-dropout, Deep Ensemble modelling and Test-time augmentation) have been explored in the context of image segmentation (for RT) in various studies. For an overview of these applications, see Table 3.4, and for a detailed explanation, see Appendix section 9.1. These studies primarily focus on applying one or more methods and/or metrics for computing uncertainty to assess the quality of an autosegmentation algorithm. While most research has identified a correlation between erroneous autosegmentation and uncertainty, there has been limited exploration into understanding the behaviour of uncertainty itself in a real-world use-case. However a deeper understanding of uncertainty behaviour is essential for reliable and safe application of uncertainty in a clinical workflow for QA purposes. Furthermore, gaining insight into uncertainty behaviour, could aid to standardise research efforts in RT by providing a more unified approach to method and metric selection based on the purpose of uncertainty, rather than relying on disparate methods and metrics across studies.

The remainder of this thesis will focus on implementing and validating various uncertainty computation techniques, as well as examining their behaviour in practical scenarios.

Table 3.4: Overview of applications for uncertainty computation in existing literature on medical image segmentation

Study	MC-dropout	Deep Ensemble	Aleatoric uncertainty	Research aim
Jungo et al. [77]	X			Relation between doubt score (based on uncertainty and distance to lesion) and dice score
de Vries et al. [9]	X		X	Relation between uncertainty estimation and jaccard index
Nair et al. [71]	X			Compared logit output variance, sample variance, predictive entropy, and mutual information
Roy et al. [78]	X			Relation between MC samples overlap as uncertainty measure and dice score
Wang et al.	X		X	Compared MC-dropout as epistemic uncertainty with test-time augmentation as aleatoric uncertainty for dice score and average symmetric surface distance improvement
Leousi [42]	X	X	X	Compared uncertainty through various methods (MC-dropout, deep ensemble, and logit obtained variance) with accuracy, precision, DSC, Hausdorff distance, and average surface distance
Jungo et al. [79]	X	X	X	Evaluated several uncertainty estimation methods for brain tumor segmentation in terms of calibration, segmentation error localization, and segmentation failure detection
van Rooij et al. [80]	X			Applied MC-dropout and MC-dropout with added Gaussian noise to the model parameters and compared uncertainty to autosegmentation quality
Bhat et al.	X	X	X	Compared the influence of several uncertainty methods (MC-dropout, deep ensemble, test-time augmentation) on liver lesion autosegmentation to shape-based features to classify autosegmentation quality (acceptable/inacceptable)
Sahlsten et al. [81]	X	X		Evaluated applicability of uncertainty computation (MC-dropout and deep ensemble) for autosegmentation quality performance assessment (DSC, mean surface distance, and Hausdorff distance)
Outeiral et al. [82]				Compared the autosegmentation NN output before binarization (i.e., Softmax scores) as a score map to autosegmentation quality (DSC, Hausdorff distance, mean surface dice, and surface dice) in cervical cancer patients

Chapter 4

Head and Neck Cancer Organs at Risk Autosegmentation

4.1 Introduction

Autosegmentation models have been implemented in the radiotherapy workflow. However, time-gain and reproducible quality by minimising inter- and intra-observer variability have been limited as still many adjustments are being made. [11]. For optimal application of autosegmentation and a possible future role in the fast-paced adaptive RT, a method for automated QA has to be sought. [25, 7, 83] An interesting emerging area for guiding clinical implementation of AI in radiation oncology, is uncertainty computation. [33] However, to compute uncertainty on NNs, for example with MC-dropout, it is necessary to perform adjustments to the model structure. The currently used autosegmentation model for HNC OARs at the department of radiotherapy at the University Medical Centre Groningen (UMCG) is integrated into the Mirada framework (WorkflowBox 2.0, Mirada Medical Ltd., Oxford, United Kingdom). It is inaccessible for adjustments to the network structure. Therefore, it is necessary to develop a custom autosegmentation network.

Over the years, automated segmentation has shifted from Atlas-based to Deep Learning-based algorithms, with improvements in performance, time efficiency and reduction of inter- and intra-observer variability. [57] A contemporary traditional choice for medical autosegmentation is a CNN-based approach. For CNN-based autosegmentation the state-of-the-art nnU-Net [43] provides an out-of-the-box framework that processes the data and provides a configuration of the network by a set of fixed and flexible parameters, automatically determined on the data fingerprint.

This study aims to train an autosegmentation network for 19 OARs for HNC patients receiving radiotherapy. The objective is to achieve performance that is at least comparable to the currently clinically used autosegmentation model and the standard performance of HNC OAR autosegmentation models in literature. The network will be compared both on geometric and dosimetric performance. In the following part of the research, this model will be used to implement uncertainty computation and as a reference for baseline model performance.

4.2 Method

4.2.1 Dataset and preprocessing

The original Mirada model was trained on UMCG data in a previous study by van Dijk et al. [6]. This data archive was still available and complemented with new data. A total dataset of 844 HNC patients was obtained who received RT with or without systemic treatment between July 2007 and March 2017. For all patients, we checked that a planning computed tomography (CT) scan and an RTSTRUCT file was available. Additionally, the patient was excluded if the voxel sizes were larger than 1.5x1.5x4.0mm or if the CT scan and RTSTRUCT file did not match. For a flow diagram of the inclusion and exclusion of patients, see Figure 4.1. Applying all exclusion criteria resulted in a final set of 818 patients.

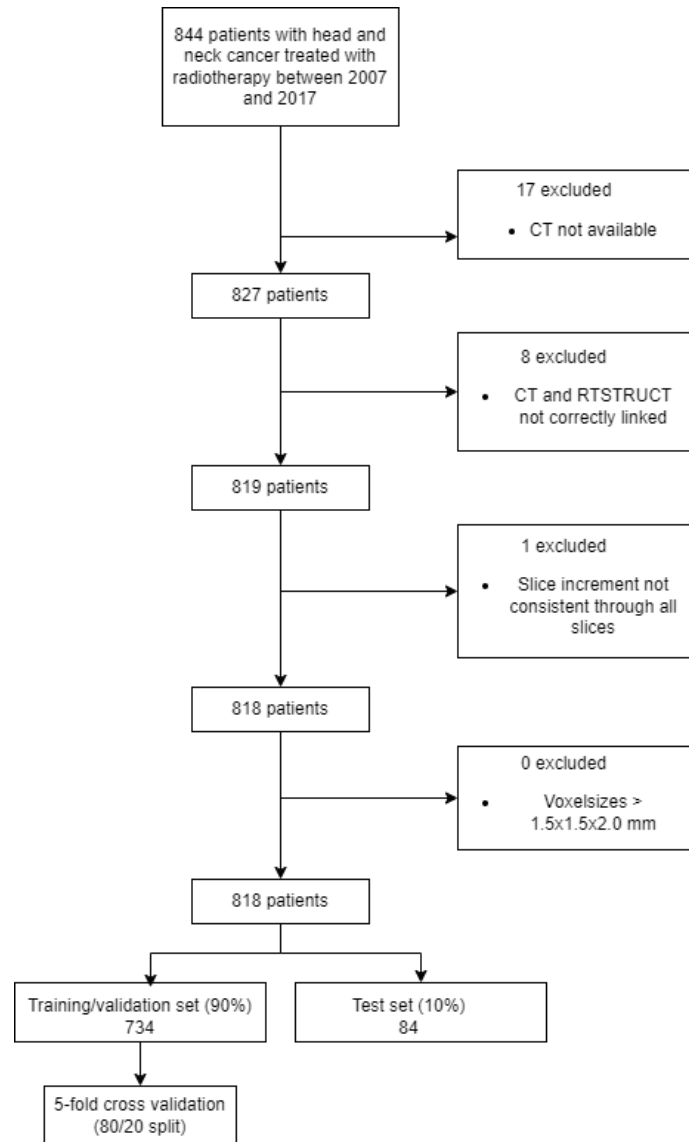


Figure 4.1: Flowchart of patients after applying exclusion and inclusion criteria

For all patients, we analysed the availability of 19 OARs in the radiotherapy structure set (RTSTRUCT) as well as the CT information. In 12 patients the left or right submandibular gland was missing, as it was removed during a combined neck dissection and submandibular gland extirpation to remove lymph nodes. These patients were not excluded from the

dataset. The CT information regarding CT-spacing and dimensions can be found in ???. Most CT scans had a resolution of 0.9766x0.9766x2.0 mm with some outliers with a pixel spacing of 0.64x0.64mm or 1.60x1.60mm. Slice thickness for 28 patients was 4.0mm, for 11 patients 1.0 mm and for the remaining 779 patients it was 2.0 mm. CT-dimensions in the axial plane were 512x512 for all patients, and the number of z-slices ranged between 48 and 466 with a mean of 173.

The dataset was preprocessed for training the DL network by extracting the relevant structures from the RTSTRUCT-file. For fast loading and processing, the data (i.e. CT-scan and RTSTRUCT-file) were converted to Nifti-format using the dicom2nifti package version 2.4.10 (Icometrix NV) in Python version 3.11.7. Further preprocessing for nnU-Net included standardising the CT- and RTSTRUCT filenames and folder structure.

4.2.2 nnU-Net preprocessing

No further manual preprocessing of the data or model training parameter optimisation was necessary before training, as the nnU-Net is a self-configuring network and executed without manual intervention on model or data hyperparameters. [43]. The network includes some fixed parameters as well as variable parameters based on dataset features. Fixed parameters include features such as the training scheme: always 1000 epochs and a high starting learning rate of 0.01 which decreases using the polyLR schedule. The network is always optimised using stochastic gradient descent and uses an average of the dice loss and cross-entropy loss as the loss function. Furthermore, the basic network architecture is not subject to data characteristics. It is a U-Net with two computational blocks per resolution stage consisting of plain convolutions followed by instance normalisation and a Leaky ReLU layer.

Based on dataset characteristics the network architecture has to be adapted to accommodate the size and spacing of the input data. This includes, among other parameters, the depth of the network, input patch size, batch size. Furthermore, image preprocessing techniques such as intensity normalisation and image resampling are applied based on the data fingerprint [43] For a full overview of the model and preprocessing hyperparameters, see Appendix 9.2.

4.2.3 Model training

The nnU-Net model (pulled from <https://github.com/MIC-DKFZ/nnU-Net>, 20-12-2023) [43] was trained using the Hábrók high-performance computer cluster from the Centre for Information Technology of the University of Groningen, equipped with two Nvidia V100 GPU accelerator cards both with 128GB memory (Python 3.9.6, Pytorch 2.1.2 with CUDA 12.1.1). The model was developed using 5-fold cross-validation on 734 patients. Data augmentation during training was performed, however the nnU-Net trainer without mirroring was utilised, as there are multiple bilateral structures in the OAR set. For a description on the specific commands and file structure that was used to develop the nnU-Net autosegmentation network, see Appendix section 9.3.

4.2.4 Geometric performance

Model performance was evaluated on 84 patients, randomly selected from the complete patient dataset. Performance was split into two categories: geometric and dosimetric autosegmentation quality. Firstly, the geometric performance assessment approach will be explained. Geometric performance was assessed using the Dice-Sørensen coefficient (DSC) score, Hausdorff distance (HD), HD 95th percentile, added path length (APL) and surface DSC score. For an illustration of all metrics, see Figure 4.2

Dice Similarity Coefficient (DSC)

The similarity between the DL contour and clinical contour was measured with the DSC score, defined as in Equation 4.1. For boolean data, such as segmentation structures, the equation can then be rewritten using the number of true positive (TP), true negative (TN) and false positive (FP) voxels.

$$DSC(\hat{y}, y^*) = \frac{2|y^* \cap \hat{y}|}{|y^*| + |\hat{y}|} = \frac{2TP}{2TP + FP + FN} \quad (4.1)$$

With $\hat{y}$ the predicted DL contour, y^* the true clinical contour.

The DSC score between each DL contour and the clinical contour (CC) was computed using a custom-developed Python script using Numpy (1.26.2).

Hausdorff Distance

The HD returns the maximum distance of a point on one contour to the closest point on the other contour. [84] The HD was computed using the Segmentation Metrics Package in Python (version 1.1.6) [85]. This package also includes an option to compute the 95th percentile of the distances between the DL contour and the CC (95% HD). This eliminates the effect of outliers.

Surface Dice Similarity Coefficient

The Surface DSC was introduced with a similar purpose as the APL; to provide a more clinically relevant measure that indicates the adaption necessary to obtain the CC from a DL contour. The metric is defined as the intersection surface of two contours, normalised by the union of the two contour surfaces. As opposed to the volumetric DSC score which evaluates a whole volume, the surface DSC score focuses on the surfaces. The value for the surface DSC defines what portion of the surface of a contour was accurate. For example if the surface DSC score is 80%, this means that approximately 80% of the contour is correct and that 20% needs to be adjusted. [86]. The surface DSC was computed using the Surface distance metrics package in Python. [87]

Added Path Length

The APL measure is defined as the number of voxels added to an autosegmentation contour to result in the clinical contour, both expansion and shrinkage. This is a better indication of contour adaptation time and time-saving when using autosegmentation. This provides a more clinically focused contour evaluation. [88]. The APL was computed using the Autosegmentation Spatial Similarity Metrics package in Python. [89]

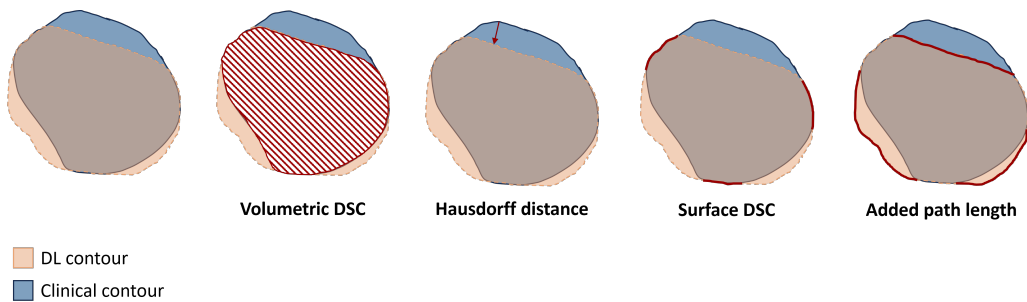


Figure 4.2: Geometric segmentation performance metrics. Figure adapted from [88]

4.2.5 Dosimetric performance

Dosimetric performance evaluation is essential alongside geometric performance evaluation, as there is often a discrepancy between the two and clinical impact is thus not adequately captured by sole geometric performance evaluation. [90, 8] The quality of an autosegmentation can be geometrically adequate, but still lead to substantial mean dose differences and thus have a significant clinical impact.

Dosimetric performance was assessed using various dosimetric measures: mean, D95 and maximum dose differences between the DL contour and the CC. Where the D95 is the dose that covered 95% of the structure volume. The dose differences between the DL contour and CC were computed on the original clinical treatment plan, based on the CC set. See Equation 4.2

$$dose_difference = D_{CC}(CC) - D_{CC}(DLC) \quad (4.2)$$

With $D_{CC}(X)$ dosimetric endpoint of interest (e.g. mean dose) of the treatment plan based on the CC set evaluated on structure set X.

Treatment plans comprised 3D Conformal Radiation Therapy (CRT), Simultaneous integrated boost (SIB) 7 Field IMRT, or 2-ARC Volumetric-Modulated Arc Therapy (VMAT). Most patients were prescribed 70Gy to the planning target volume (PTV) in 35 fractions such that 98% of the elective PTV receives at least 95% of the prescribed dose and a maximum of 4% receives 107 % of the prescribed dose. Dose to OARs was minimised using constraints according to the HNC radiotherapy protocol.

It is important to note that various definitions of dosimetric differences, evaluate distinct clinical impact of a contour set. Contouring is namely used for multiple purposes in RT. Firstly, contours are used to make a treatment plan that minimises dose to the OARs and maximises target coverage. Secondly, contours are used to assess the quality of a treatment plan by evaluating the mean and max dose on OARs. In this research, the impact of autosegmentation network on the dosimetric performance was evaluated by comparing the difference between the DL set and the CC on the clinical treatment plan: an approximation of the effect of autosegmentation on treatment plan evaluation. [8] This is also the metric that was obtained by Van Dijk et al. [39] and was, therefore, best suited for this research where a similar performing model was pursued. For more explanation on the difference between various approaches to compute dosimetric differences between contours and their clinical impact, please refer to Appendix section 9.5

4.2.6 Performance evaluation

Model performance was compared with geometric and dosimetric performance in the previous study by van Dijk et al. [6] and with other published HNC OAR segmentation models. [80, 4, 58]

Table 4.1: 5-fold Cross-validation performance. Mean DSC score over all OARs

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
mean validation DSC scores	0.85	0.85	0.85	0.85	0.85

DSC = Dice-Sørensen coefficient, OAR = organ at risk

4.3 Results

The nnU-Net was trained with 5-fold cross-validation leading to a training set of approximately 589 patients and a validation set of 147 patients per fold with a possible variation of ± 2 patients between the folds. Training and validation of one fold took about 15 hours. The 3D full-resolution version of the nnU-Net was trained, with batch size 2, a patch size of $[64, 160, 160]$ (Z, X, Y), spacing of $[2.0 \text{ mm}, 0.9765625 \text{ mm}, 0.9765625 \text{ mm}]$ (Z, X, Y). A full list of the network configuration and parameter settings can be found in Appendix section 9.2. For a visualisation of one of the training folds with loss and validation DSC scores, see Figure 4.3. For the mean validation DSC scores per fold, see Table 4.1

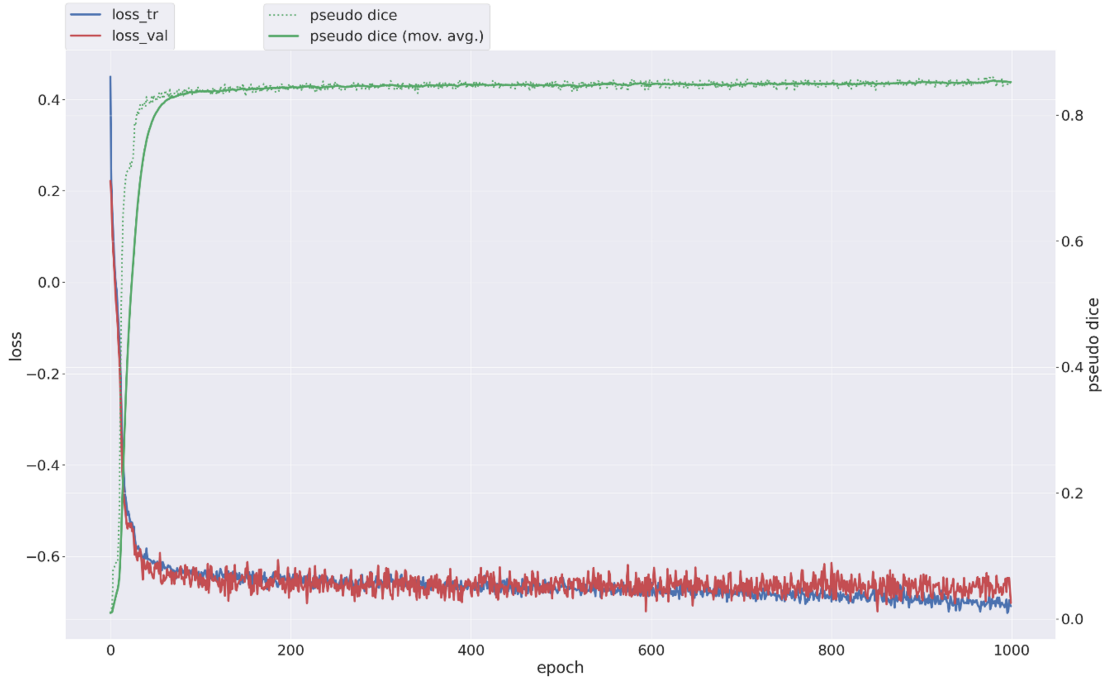


Figure 4.3: One of the training folds for nnU-Net with loss and mean validation DSC scores over all OARs.

Figure 4.4 shows an example of an autosegmentation of the 20 HNC OARs where the clinical contour set is projected as the ground truth alongside the DL predicted autosegmentation obtained with the nnU-Net for one test patient on various slices.

Geometric performance of the autosegmentation network on the test set can be found in Table 5.1. For the DSC score, box plots per structure containing all patients can be found in Figure 4.5. Box plots of the HD 95th percentile, APL and surface DSC can be found in Appendix section 9.6. A comparison of the obtained volumetric DSC score with existing literature can be found in Table 4.4

Dosimetric performance of the autosegmentation network on the test set can be found in Table 5.2. Box plots of the mean dose differences can be found in Figure 9.8. Box plots of the max and D95 dose differences can be found in Appendix section 9.7.

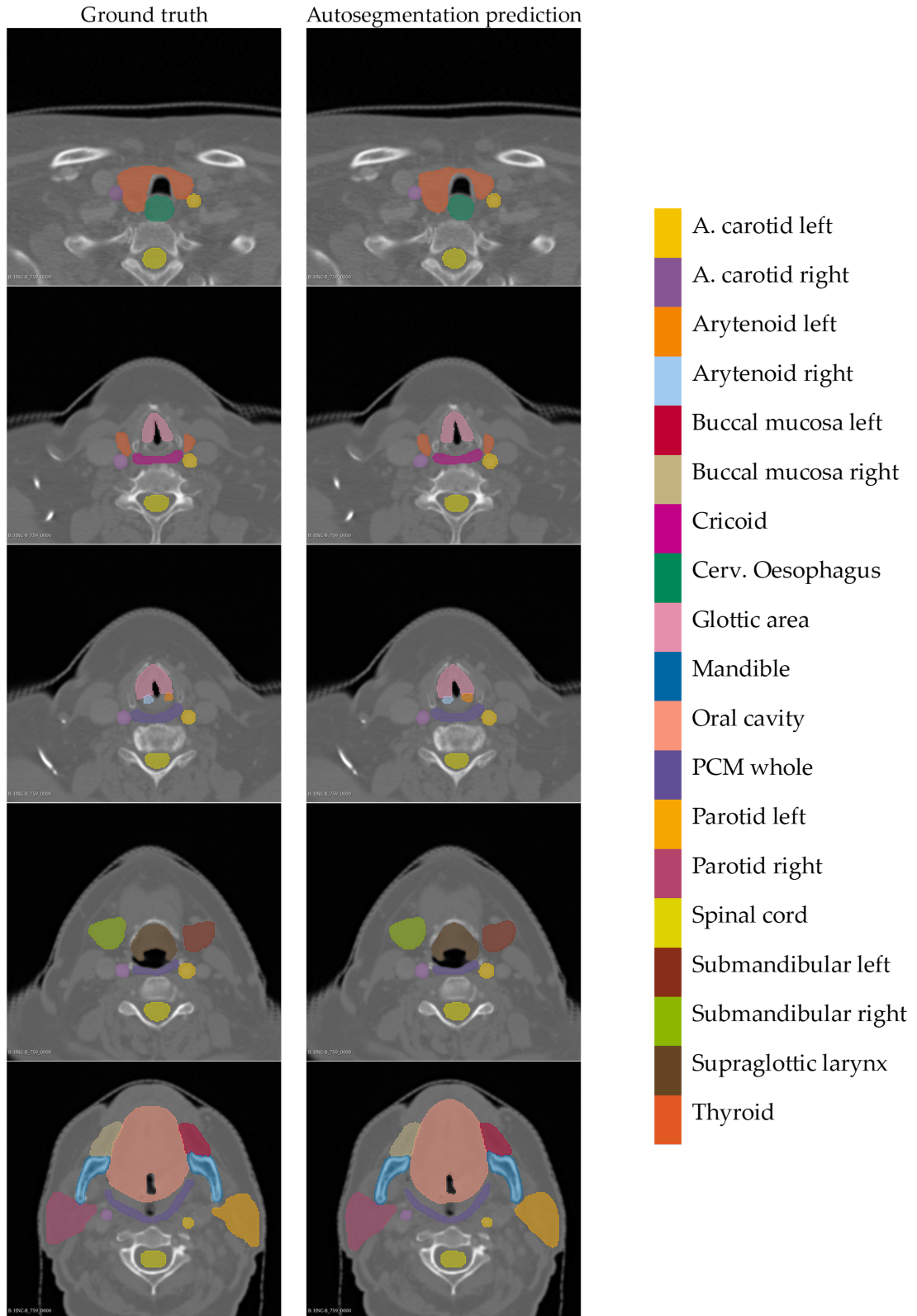


Figure 4.4: Ground truth clinical contour set and DL predicted contour set obtained with the nnU-Net on test patient 759.

Cerv. = cervical, PCM = pharyngeal constrictor muscle

Table 4.2: Geometric performance evaluation: average for all test patients

OAR	Vol. DSC $\uparrow$ mean $\pm$ std	HD[mm] $\downarrow$ mean $\pm$ std	HD95[mm] $\downarrow$ mean $\pm$ std	Surf. DSC $\uparrow$ mean $\pm$ std	APL [voxels] $\downarrow$ mean $\pm$ std
A. carotid L	0.88 $\pm$ 0.04	5.06 $\pm$ 4.42	1.60 $\pm$ 2.20	0.93 $\pm$ 0.07	1238.75 $\pm$ 463.45
A. carotid R	0.87 $\pm$ 0.04	5.15 $\pm$ 3.00	1.27 $\pm$ 0.72	0.93 $\pm$ 0.07	1103.00 $\pm$ 408.51
Arytenoid L	0.77 $\pm$ 0.14	2.25 $\pm$ 1.30	1.37 $\pm$ 1.03	0.88 $\pm$ 0.14	58.36 $\pm$ 37.88
Arytenoid R	0.76 $\pm$ 0.13	2.19 $\pm$ 1.14	1.32 $\pm$ 0.64	0.87 $\pm$ 0.13	62.07 $\pm$ 32.68
Buccal mucosa L	0.83 $\pm$ 0.04	6.95 $\pm$ 2.44	2.36 $\pm$ 1.04	0.76 $\pm$ 0.10	996.90 $\pm$ 315.62
Buccal mucosa R	0.83 $\pm$ 0.04	6.66 $\pm$ 2.69	2.28 $\pm$ 1.07	0.77 $\pm$ 0.09	1023.13 $\pm$ 345.18
Cricoid	0.83 $\pm$ 0.07	3.52 $\pm$ 2.53	1.67 $\pm$ 1.71	0.82 $\pm$ 0.11	898.92 $\pm$ 270.25
Cerv. oesophagus	0.80 $\pm$ 0.12	3.59 $\pm$ 2.60	1.92 $\pm$ 1.73	0.76 $\pm$ 0.15	379.95 $\pm$ 211.82
Glottic area	0.84 $\pm$ 0.08	3.40 $\pm$ 1.33	1.29 $\pm$ 0.80	0.88 $\pm$ 0.09	556.26 $\pm$ 314.92
Mandible	0.96 $\pm$ 0.01	3.80 $\pm$ 4.91	1.00 $\pm$ 0.00	0.97 $\pm$ 0.03	2417.87 $\pm$ 1116.01
Oral cavity	0.94 $\pm$ 0.01	6.35 $\pm$ 2.56	1.83 $\pm$ 0.60	0.81 $\pm$ 0.07	4902.50 $\pm$ 1265.06
PCM whole	0.80 $\pm$ 0.05	8.02 $\pm$ 3.22	2.03 $\pm$ 1.21	0.83 $\pm$ 0.09	3093.83 $\pm$ 1006.55
Parotid L	0.88 $\pm$ 0.04	11.65 $\pm$ 7.87	2.66 $\pm$ 1.93	0.77 $\pm$ 0.09	2346.02 $\pm$ 916.27
Parotid R	0.88 $\pm$ 0.04	10.63 $\pm$ 7.60	2.49 $\pm$ 1.34	0.78 $\pm$ 0.09	2269.33 $\pm$ 690.46
Spinal cord	0.94 $\pm$ 0.03	3.46 $\pm$ 2.38	1.16 $\pm$ 0.53	0.94 $\pm$ 0.06	1008.46 $\pm$ 523.79
Submandibular L	0.89 $\pm$ 0.05	4.05 $\pm$ 1.42	1.64 $\pm$ 0.95	0.86 $\pm$ 0.09	780.98 $\pm$ 335.59
Submandibular R	0.87 $\pm$ 0.12	5.10 $\pm$ 4.46	2.40 $\pm$ 4.22	0.84 $\pm$ 0.14	789.86 $\pm$ 322.02
Supra. larynx	0.87 $\pm$ 0.04	4.50 $\pm$ 1.91	1.59 $\pm$ 0.85	0.85 $\pm$ 0.09	1338.16 $\pm$ 546.33
Thyroid	0.89 $\pm$ 0.07	5.36 $\pm$ 4.48	1.73 $\pm$ 2.88	0.88 $\pm$ 0.11	1841.23 $\pm$ 1663.90

OAR = organ at risk, Vol. = volumetric, DSC = Dice-Sørensen coefficient, HD = Hausdorff distance, HD 95th = 95th percentile of Hausdorff Distance, Surf. = surface, APL = added path length, std = standard deviation, L = left, R = right, Supra. = supraglottic, Cerv. = cervical.

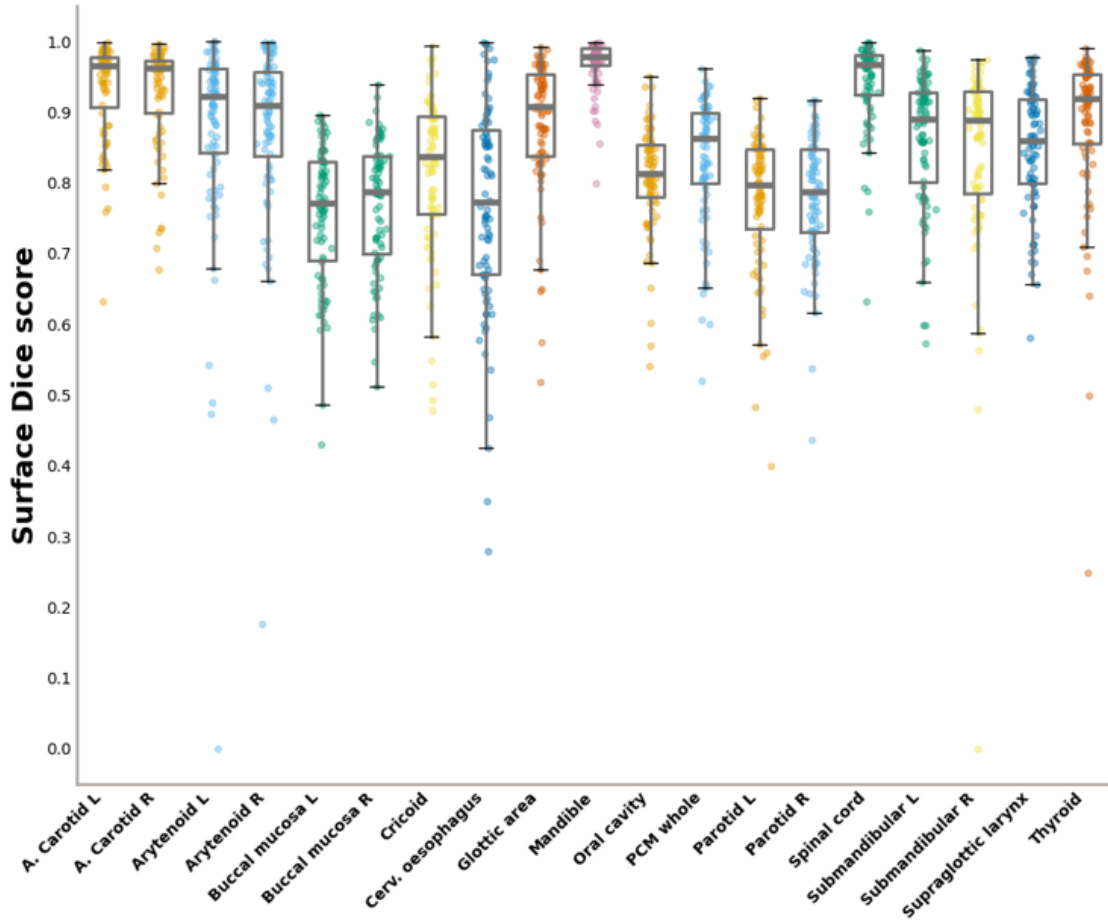


Figure 4.5: Box plots of surface DSC score per OAR for all test patients on nnU-Net autosegmentation model.

Table 4.3: Dosimetric performance evaluation: average for all test patients

OAR	Abs. Mean dose diff. mean $\pm$ std[Gy] ↓	Abs. Max dose diff. mean $\pm$ std[Gy] ↓	Abs. D95 dose diff. mean $\pm$ std[Gy] ↓
A. carotid L	1.00 $\pm$ 0.78	0.21 $\pm$ 0.37	0.20 $\pm$ 0.33
A. carotid R	1.29 $\pm$ 1.12	0.26 $\pm$ 0.44	0.24 $\pm$ 0.56
Arytenoid L	0.40 $\pm$ 0.59	0.41 $\pm$ 0.58	0.32 $\pm$ 0.41
Arytenoid R	0.41 $\pm$ 0.59	0.59 $\pm$ 0.89	0.42 $\pm$ 0.64
Buccal mucosa L	0.67 $\pm$ 0.79	0.93 $\pm$ 1.43	0.68 $\pm$ 1.09
Buccal mucosa R	0.68 $\pm$ 0.83	0.97 $\pm$ 1.66	0.74 $\pm$ 1.40
Cricoid	1.10 $\pm$ 1.70	0.66 $\pm$ 1.08	0.59 $\pm$ 0.89
Cerv. oesophagus	1.17 $\pm$ 1.65	2.34 $\pm$ 4.51	2.11 $\pm$ 3.66
Glottic area	0.37 $\pm$ 0.88	0.40 $\pm$ 0.63	0.29 $\pm$ 0.54
Mandible	0.16 $\pm$ 0.16	0.18 $\pm$ 0.30	0.10 $\pm$ 0.18
Oral cavity	0.34 $\pm$ 0.37	0.34 $\pm$ 0.87	0.24 $\pm$ 0.41
PCM whole	0.78 $\pm$ 0.77	0.41 $\pm$ 1.40	0.26 $\pm$ 0.64
Parotid L	1.27 $\pm$ 1.32	0.83 $\pm$ 2.14	1.03 $\pm$ 1.83
Parotid R	0.92 $\pm$ 1.04	0.34 $\pm$ 0.54	0.71 $\pm$ 1.36
Spinal cord	0.38 $\pm$ 0.61	0.34 $\pm$ 0.53	0.12 $\pm$ 0.15
Submandibular L	0.25 $\pm$ 0.31	0.33 $\pm$ 0.83	0.38 $\pm$ 0.80
Submandibular R	0.43 $\pm$ 0.80	0.31 $\pm$ 0.67	0.42 $\pm$ 1.24
Supra. larynx	0.77 $\pm$ 1.12	0.31 $\pm$ 0.59	0.33 $\pm$ 0.70
Thyroid	0.56 $\pm$ 1.36	0.47 $\pm$ 0.89	0.43 $\pm$ 0.94

OAR = organ at risk, Abs. = absolute, diff. = difference L = left, R = right, Cerv. = cervical, PCM = pharyngeal constrictor muscle, Supra. = supraglottic,

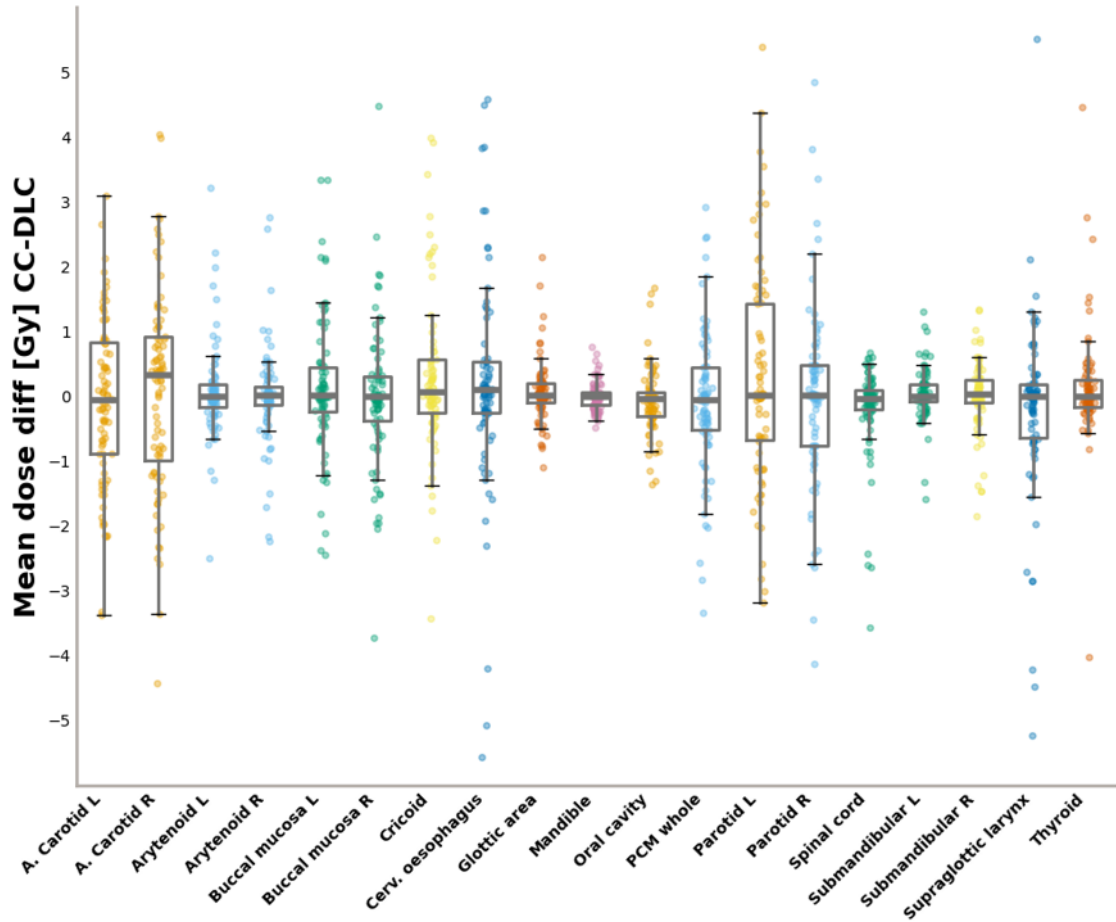


Figure 4.6: Box plots of mean dose difference per OAR for all test patients on nnU-Net autosegmentation model

Table 4.4: Volumetric DSC score comparison with existing literature

OAR	nnU-Net	van Dijk et al. [6]	Liu et al. [4]	van Rooij et al. [80]	Vrtovec et al. [58] review(range)
A. carotid L	0.88	0.71	-	-	0.91
A. carotid R	0.87	0.66	-	-	0.91
Arytenoid L	0.77	0.56	-	-	-
Arytenoid R	0.76	0.52	-	-	-
Buccal mucosa L	0.83	0.75	-	-	-
Buccal mucosa R	0.83	0.76	-	-	-
Cricoid	0.83	0.65	-	0.73	-
Cerv. oesophagus	0.80	0.50	-	0.60	0.35- 0.86
Glottic area	0.84	0.71	-	-	-
Mandible	0.96	0.94	0.92	-	0.78-0.96
Oral cavity	0.94	0.91	-	0.78	0.78-0.93
PCM whole	0.80	0.67	-	0.68	0.50-0.91
Parotid L	0.88	0.83	0.82	0.83	0.77-0.91
Parotid R	0.88	0.82	0.82	0.83	0.77-0.91
Spinal cord	0.94	0.86	0.83	-	0.71-0.91
Submandibular L	0.89	0.77	0.85	0.81	0.77-0.91
Submandibular R	0.87	0.78	0.85	0.82	0.77-0.91
Supra. larynx	0.87	-	-	0.78	0.71-0.91
Thyroid	0.89	-	-	-	0.57-0.92

Results in bold indicate the best-performing network per OAR OAR = organ at risk, L = left, R = right, Cerv. = cervical, PCM = pharyngeal constrictor muscle, Supra. = supraglottic,

4.4 Discussion

For the next step in improving autosegmentation implementation, uncertainty computations could be an area of interest. However, to implement uncertainty computations, an accessible autosegmentation network had to be developed as this requires network adjustments. With this study, we developed an nnU-Net autosegmentation model for 19 HNC OARs that can later be adjusted for uncertainty computations and used as a baseline model for performance reference. The model showed robust training, and high geometric performance, surpassing the current clinical model. Although dosimetric performance overall was acceptable, variations for individual patients showed a substantial dosimetric impact, highly likely to have clinical implications.

4.4.1 Model training

The identical mean validation DSC scores over all OARs per fold in Table 4.1 show stable training for all fold splits. The stable training is also observed in the loss and validation curve in Figure 4.3 where the network optimises rapidly up until around 100 epochs, from where on only slow improvement is measured and loss on the validation set stabilises. No overfitting is observed, indicating that a robust network is created.

4.4.2 Geometric performance

Geometric quality of the OAR segmentation was high, with DSC scores ranging from 0.83 to 0.96 for 17 out of 19 OARs. However, the arytenoids showed lower performance, with DSC scores of 0.77 and 0.76 for the left and right arytenoid respectively. Despite these lower DSC scores, the smallest (i.e. best) HD values were observed for the arytenoids, of 2.19 mm and 2.25 mm for the right and left arytenoid respectively, while the HD for other OARs ranged between 3.46-11.65 mm. This discrepancy between the DSC and HD for the arytenoids, can be attributed to their relatively small volume - averaging 252 and 247 voxels for the left and right arytenoid respectively - compared to larger OARs included such as the cervical oesophagus (1611 voxels) and mandible (34949 voxels).

Further confirmation of this volume-related discrepancy is evident when examining the surface DSC, a metric designed to reduce the influence of volume in geometric performance assessment for segmentation. The surface DSC for the arytenoids was notably higher (i.e. better) than their DSC scores (0.88 and 0.87 compared to 0.77 and 0.76), and relatively high compared to other OARs. This supports the known dependence of the volumetric DSC on the volume of the analysed structure. As the network is developed for a set of OARs with wide-varying volumes, the surface DSC should be used in further analyses for accurate representation of model performance.

The parotid glands demonstrated the highest (i.e. worst) HD values, with averages of 11.65 mm for the left and 10.63 mm for the right parotid gland, in comparison to the HD ranges observed for other OARs (2.19–8.02 mm). This parotid segmentation challenge is further observed in the relatively low surface DSC and when visually evaluating the autosegmentation results of the parotid glands. Errors are made especially in the anterior part of the parotid which is a notoriously difficult region to delineate. [11] This observation is further confirmed when observing the HD 95th percentile results, where the highest 5% of the HD values are excluded (possibly the anterior portion). The HD95 for the parotid glands—2.66 mm and 2.49 mm—are notably closer to the range observed for other OARs (1.00–2.40 mm), suggesting that the anterior portion is indeed the core challenge in parotid segmentation.

4.4.3 Patient level geometric performance

Figure 4.5 shows even spread in surface DSC scores between patients for most OARs, with highest variation for the cervical oesophagus. The variability can be attributed to the misidentification of feeding tubes as oesophageal tissue by the autosegmentation model. Detailed examples are provided in Appendix section 9.4. An outlier was noted for the thyroid, with one patient having a surface DSC score of 0.32, significantly lower than the range of the other patients (0.75-0.96). This drop in performance appears to be due to the model relying on the carotid artery to guide thyroid segmentation. In this case, fluid accumulation displaced the artery, leading the model to extend the thyroid segmentation incorrectly, for more details, see section 9.4

For bilateral OARs like the parotid and submandibular glands, disabling mirroring during data augmentation prevented lateral flipping of these structures, which improved segmentation accuracy. Disabling mirroring also made training more robust with a more stable decrease in the cost function and a more consistent increase in validation DSC score. For a comparison, see Appendix 9.8.

4.4.4 Geometric performance comparison with existing autosegmentation models

Van Dijk et al. [6] and Liu et al. [4] obtained inferior DSC performance for all OARs with their developed CNN. Superior performance of our model could possibly be attributed to the high quality of the automated hyperparameter tuning based on data preprocessing by nnU-Net. Another explanation could be the extension of the dataset from 693 to 818 patients compared to the model by van Dijk et al. [6]. Van Rooij et al. [80] also observed lower DSC scores, which could be due to their smaller dataset (on average 98 patients for training) and their use of individual models for each OAR. This approach, unlike the multi-OAR model developed in this study, may limit the inclusion of shared features between OARs, potentially weakening performance.

Vrtovec et al. [58] reviewed autosegmentation models for OARs in the head and neck region, and the nnU-Net results from this review generally fall at the high end of their

reported performance ranges. For most OARs, our DSC scores are on the higher end or exceed those in the literature, though slightly lower scores were observed for the carotid arteries (0.87 and 0.88). Notably, Vrtovec et al. only analysed two studies on carotid artery autosegmentation, making direct comparison more challenging.

Overall, the geometric performance of the nnU-Net model in this study is robust, showing improvements over previously developed models and meeting the high standards expected within the field of radiotherapy. However, some individual patients exhibited inferior performance, possibly having a clinical impact through dosimetric differences, that will now be explored.

4.4.5 Dosimetric performance

Overall, the dosimetric evaluation showed that dose differences for most OARs are generally acceptable, with average mean dose differences falling below 1.0 Gy—a threshold commonly used to assess dosimetric impact. However, certain OARs, including the left and right carotid arteries, cricoid, cervical oesophagus, and left parotid, exhibited higher mean dose differences (1.00 Gy to 1.29 Gy), indicating potential dosimetric concerns that should be addressed before clinical implementation.

Analysis of individual patient data, shown in Figure 9.8, reveals that while average dose discrepancies are minimal and evenly distributed between underdosage and overdosage, substantial variation exists between patients, with absolute mean dose differences reaching up to 4-5 Gy per OAR. For example, significant variability was noted for the left parotid gland and cervical oesophagus. Specifically, high dose differences for the cervical oesophagus were associated with patients having a larynx tumour or neck irradiation and a low DSC score (around 0.70), attributed to inadequate cranial and caudal extension of the structure. For the parotid glands, large dose differences were particularly observed in cases of oropharyngeal cancer patients where the anterior part of the parotid gland was absent in the DL contour. For these patients the tumour is in proximity of the parotid gland, amplifying dose changes due to high dose gradients. For more details, see Appendix section 9.4.

This variability in dosimetric quality highlights that while average metrics are generally favourable, specific patient cases reveal substantial dose differences that could impact treatment planning, plan evaluation, model-based patient selection and ultimately radiation toxicity in patients. Comparison with the study by van Dijk et al. [6] shows that their results had larger average dose differences. The study by van Rooij et al. [80] cannot be directly compared due to their use of different dose analysis methodologies which included the creation of new treatment plans.

Evaluating Assessing the (surface) DSC score in relation to the treatment plan is essential to help identify patients needing manual adjustments to prevent clinical impact of autosegmentation. Currently, often solely geometric evaluation is performed in QA, however, this may overlook segmentation outputs requiring clinically necessary correction.

4.4.6 Limitations

This study has several limitations. First, while patients with missing submandibular glands were included in the training set, none were assigned to the test set due to random partitioning. Consequently, the model's performance on these patients remains unassessed. Furthermore, it is important to note that the data set comprised a highly curated distribution with no patients receiving post-operative RT, which notoriously presents more challenging delineation cases.

Secondly, the treatment plans were manually created based on the clinical contour set. This may introduce bias, potentially affecting the dosimetric analysis. Our analysis approximated the dosimetric impact on treatment plan evaluation, but a comprehensive analysis would require metrics that assess contour quality and treatment plan development, as discussed in section 9.5. [8] Additionally, clinical impact metrics such as normal tissue complication probability (NTCP) were not computed.

Finally, the substantial variation in dose differences among patients indicates that autosegmentation models require QA during clinical implementation. Dose differences exceeding 1 Gy in some patients could significantly impact treatment outcomes. Incorporating uncertainty maps—providing voxel-level confidence—could enhance QA to manually adjust these specific cases, as fully manual QA is suboptimal. The implementation of uncertainty computation in existing models is explored in the following section.

4.5 Conclusion

This study trained and validated an nnU-Net autosegmentation model for 19 head and neck cancer OARs in RT, achieving performance at the high end of current literature. The model demonstrated high DSC scores for all OARs, providing a significant improvement over existing clinical models. This advancement offers the potential for time savings and reduced inter- and intra-observer variability in segmentation. Although dosimetric evaluations showed generally acceptable average dose differences, notable variability between patients highlights the ongoing need for QA. Integrating uncertainty computation into the model could enhance this QA process.

Chapter 5

Reliability Evaluation of Uncertainty Computation

5.1 Introduction

Segmentation models in RT QA is a field that could benefit from the additional knowledge about a model’s confidence, as performance fluctuates on individual patient level. This is a contributor to the suboptimal deployment of DL segmentation models in the clinic currently. Still, many adjustments are made, [11] with minimal but fluctuating dosimetric impact [91, 8]. This partially diminishes the time gain proposed with DL and re-introduces the inter- and intra-observer variability. Bayesian Deep Learning provides both a prediction and an uncertainty estimation during inference and could thus be of interest for autosegmentation QA. [34]

The use of uncertainty for autosegmentation QA or DL model improvement is becoming an increasingly popular area of research in RT. Uncertainty computation has been applied for various purposes, focusing mainly on autosegmentation failure detection, but also for model calibration or active learning. [33] Various methods exist to perform uncertainty computation, where a choice has to be made for an uncertainty computation *method* (e.g. Monte Carlo Dropout, Deep Ensembles or Test-time Augmentation) and uncertainty computation *metric* (e.g. entropy, variance or mutual-information).

Uncertainty provides an estimation of the output probability density, $p(\hat{y}|x, D)$, i.e. the confidence. Uncertainty computation is based on an approximation of the posterior of the DL model weights, $p(w|D)$, given the training data D and input x . Uncertainty is computed by generating multiple output samples which, when combined, approach the Deep Gaussian Process of the segmentation model through variational approximation. [34]. In medical image segmentation of 3D volumes, this leads to a voxel-wise uncertainty volume (i.e. uncertainty map) where the value per voxel translates to the confidence that the voxel is part of the assigned structure in the predicted segmentation. [39]

The uncertainty computation *method* defines how the various output samples of a single input are obtained. The three most popular methods in the DL medical image segmentation domain to obtain the output samples are MC-dropout [34], Deep Ensemble modelling [35] and Test-time augmentation [36, 37]. [33] The various output samples are combined with an uncertainty *metric*. The most used uncertainty metrics are variance, entropy and mutual information. [33] For a more in-depth explanation of uncertainty computation and uncertainty methods and metrics, please refer to Chapter 3.

Most studies investigating uncertainty application in RT focus on a single uncertainty computation *method* and *metric*. Consensus on what uncertainty method and metric to use is lacking [33, 92] Whitbread et al. [92] found that MC-dropout and Test-time Augmentation produce distinct uncertainty maps, with significant effects from hyper-parameter settings like dropout rate. However, they only evaluated entropy as an uncertainty metric and did not examine uncertainty calibration. Jungo et al. [79] provided an in-depth analysis of MC-dropout, Deep Ensemble modelling and Test-time Augmentation for tumour segmentation and evaluated the uncertainty maps on calibration, but they did not explore other uncertainty metrics than entropy or a range of hyperparameters for MC-dropout. Camarasa et al. [93] applied various uncertainty metrics to MC-dropout and found entropy and mutual information to outperform variance.

A comprehensive evaluation of the quality and robustness of different uncertainty computation methods combined with metrics is lacking for OAR autosegmentation in RT. Furthermore, before proceeding to the next part of this thesis, the empirical uncertainty experiments for ODD, it is necessary to assess the quality of the uncertainty estimations first. Therefore, this section presents a quantitative evaluation of the three most popular uncertainty methods (MC-dropout, Deep Ensemble modelling, and Test-time augmentation)[33] and the top-performing uncertainty metrics (entropy, variance, and mutual information) [93, 33] for multi-class and class-wise uncertainty map computation.

The main objective is to establish the combination(s) of uncertainty method and metric that provide(s) reliable uncertainty maps and to create a benchmark for uncertainty evaluation in the successive section where data experiments will be performed to influence uncertainty computation. We compare the quality of the uncertainty maps considering the influence on segmentation performance and uncertainty value calibration (i.e. the relationship between confidence and accuracy). First, a brief background on the different uncertainty methods and metrics will be provided. After this, the methods and results of the experiments to assess uncertainty map quality will be explained, followed by a discussion and conclusion.

5.2 Background

5.2.1 Obtaining an output distribution: uncertainty methods

MC-dropout utilises dropout during test time to obtain the posterior of the network's weight by randomly dropping part of the network weights. Dropout is conventionally used during model training to make a network more robust and less susceptible to noise. However, when dropout is activated during inference, the model becomes a Bayesian approximator. [34]

With Deep Ensemble modelling, several models are independently trained to form an ensemble of neural networks which are individual valid variants of the DL model. When providing the networks with the same input, they each produce a variation of the output. Deep Ensemble modelling is developed to improve model performance and reliability, however, when combined they approach a Bayesian neural network when a proper scoring rule is used. [35].

Similarly to MC-dropout, Test-time augmentation was originally proposed for network training. It is a method to enlarge a limited dataset by adding random transformations during training to input samples, such as rotation, intensity-shifting, flipping and shifting. However, when used during inference, it obtains a set of variations of an output sample by performing the same random augmentations used when training the network. [36, 37]

Specifically for MC-dropout, hyperparameter tuning of the dropout location and dropout rate determines the quality of the uncertainty map. [92, 93, 94]. The dropout location determines which NN layers dropout is applied on. The dropout rate determines the chance of a weight in the network becoming zero and thus specifies how much of a network is dropped.

All three methods lead to a collection of output samples which involve stochastic variations, through variations in the input in the case of Test-time augmentation or the network architecture in the case of MC-dropout and Deep Ensemble modelling.

5.2.2 From samples to uncertainty map: uncertainty metrics

Uncertainty metrics translate a distribution of output samples to an uncertainty map. All output samples together form an array of size $T \times C \times X \times Y \times Z$ with T the number of samples, C the number of classes (in case of one-hot encoding, otherwise C is 1), and $X \times Y \times Z$ the CT volume size. Different statistical methods exist in information theory to concatenate the information of these probabilistic samples into an uncertainty map of size $X \times Y \times Z$ (i.e. uncertainty metrics). Nair et al. [71] provided an in-depth discussion and derivation of the commonly used uncertainty metrics: predictive entropy, mutual information and variance.

Each of these metrics theoretically captures a different type of uncertainty. The (predictive) entropy is believed to encompass both epistemic and aleatoric uncertainty, where mutual information is suggested to represent solely the epistemic uncertainty. [95] An in-depth background on these measures and their theoretical/mathematical differences can be found in section Chapter 3. In practice, entropy is most commonly used as an uncertainty metric for MC-dropout and variance for Deep ensemble modelling, however, there is no consensus. [33, 39, 79, 42, 9, 96, 97] Some also use mutual information [98, 96], where others (also) compute the variance to obtain MC-dropout uncertainty maps[99, 96]. Formal definitions of the uncertainty metrics are:

$$Variance(p) = \hat{\sigma}^2 = \frac{1}{T} \sum_{i=1}^T (\hat{p}_{ij} - \hat{p}_j)^2 \quad (5.1)$$

$$Entropy = H(p) = - \sum_{c=0}^C \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right) \log \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right) \quad (5.2)$$

$$MI = \mathcal{M} = -H(p) + \frac{1}{T} \sum_c \sum_t p(\hat{y} = c | x_i^*, W_t) \log(p(\hat{y} = c | x_i^*, W_t)) \quad (5.3)$$

With MI mutual information, C the number of classes, T the number of Monte Carlo samples, $\hat{p}_j$ the mean of all samples for class j and $\hat{p}_{ij}$ the i^{th} sample for class j. $\hat{y}$ the predicted output class, x_i^* the test input.

When the segmentation output consists of more than one class, uncertainty can be computed for each class individually (classwise) as well as by combining (i.e. averaging) the uncertainty over all classes (multi-class). The equations above describe a multiclass approach. With a classwise evaluation, the uncertainty function is applied to the channels for the segmentation structure under study against the other labels (i.e. one-versus-all). This leads to C uncertainty maps of size $X \times Y \times Z$. [93] A formal definition for example class $C = 0$ out of N for classwise entropy:

$$\begin{aligned}
 \text{Classwise_entropy} = & \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = 0 | x_i^*, W_t) \right) \log \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = 0 | x_i^*, W_t) \right) \cdot \\
 & \sum_{c=1}^N \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right) \log \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right)
 \end{aligned} \tag{5.4}$$

5.3 Methods

The quality of uncertainty is mainly determined by expected behaviour, as no ground truth for uncertainty exists. [67] Two important concepts indicating the trustworthiness of uncertainty maps are the stability in model accuracy performance and the reliability of uncertainty values. This section will outline the theoretical foundation and practical approach to assess uncertainty in these contexts.

Autosegmentation performance evaluation

First and foremost, it is important to ensure that the developed network results in acceptable autosegmentation predictions and that autosegmentation performance is not significantly compromised when implementing uncertainty computations. Especially when applying MC-dropout, the possibility arises that the obtained autosegmentation is altered considerably when dropping an essential part of the network structure. [93]

Evaluating the quality of autosegmentation can be done on a geometric and dosimetric level. While the former evaluates the contour quality, the latter evaluates the (clinical) impact of on the planned delivered dose. We used the same evaluation metrics as in Chapter 4. For the geometric evaluation the volumetric DSC, APL, HD, 95th percentile HD and surface DSC score were utilised. For the dosimetric evaluation, the mean and max dose differences were obtained.

5.3.1 Theory

Uncertainty map reliability evaluation

Uncertainty maps are only valuable if the uncertainty values are well calibrated: if generated probabilities (i.e. certainty) represent the likelihood of being correct (i.e. accuracy). Mukhoti and Gal [67] proposed two assumptions about uncertainty computation concerning this property, which is useful for assessing uncertainty map reliability. They stated that for uncertainty computation to be useful, the uncertainty values must be inversely related to prediction accuracies, as when viewing a frequentist approach:

1. A model is expected to be correct more often when the uncertainty is low
2. A model may or may not be accurate if the uncertainty is high

Sanity checks can be performed to ensure the assumed inverse relation between confidence and frequency exists. This is essentially what is done in reliability evaluation. The expectation is that a higher fraction of accurate voxels will be found with increasing confidence. For example, if a certainty of 80% is predicted for a set of voxels, the prediction should be correct in 80 out of 100 times. [100] Popular methods for calibration assessment are reliability diagrams [100] and the expected calibration error (ECE)-value [101].

Definition of accurate and certain

To evaluate to what extent the uncertainty values relate to accuracy, it is important to define *accurate* and *certain* itself first.

A voxel is *accurate* when the same class is predicted as it was given in the ground truth image, i.e. TP or TN. A voxel is *inaccurate* when a different class is predicted as it was given in the ground truth image, i.e. FP or false negative (FN). The ground truth in RT is the clinically used contour set, CC.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.5)$$

An uncertainty threshold, τ can be used to assign a voxel as *certain*. This leads to a dichotomisation of voxels based on uncertainty. The threshold is often the average uncertainty value over all voxels in a test set but can be any value.

Reliability diagram and ECE value

A reliability diagram plots the voxel-wise model confidence (i.e. certainty) against the observed relative frequency of positive instance (i.e. the fraction of true voxels) per bin of uncertainty values. Here, a perfectly calibrated model is represented by the line $y = x$. To compute the ECE value, the mean predicted probability in each bin and the true fraction are subtracted and summed over all bins, according to Equation 5.6. A lower ECE score denotes better calibration, with 0 representing perfect calibration.

$$ECE = \sum_{i=1}^N P(i) |o_i - e_i| \quad (5.6)$$

With o_i the true fraction of positive instances (i.e. true voxels) in bin i and e_i is the mean of the probabilities in bin i . $P(i)$ is the proportion of all instances that fall into bin i and N is the number of bins. [102]

The reliability diagram can be used to evaluate the calibration of the uncertainty maps visually. While the ECE concatenates the overall calibration performance into a single value aiding quantitative comparison between different uncertainty models.

A problem with analysing the accuracy per voxel of the whole CT volume, is the high portion of TN values, i.e. the background. This makes the evaluation dependent on image size and background portion. To address this bias, the TN values can be left out of the calculation. An option thus is to redefine accuracy as the Jaccard index:

$$J = \frac{TP}{TP + FP + FN} \quad (5.7)$$

When applying this definition of accuracy to the ECE value computation, a new metric is defined that is inherently different from the original ECE definition. This newly designed metric is termed the ECE-label.

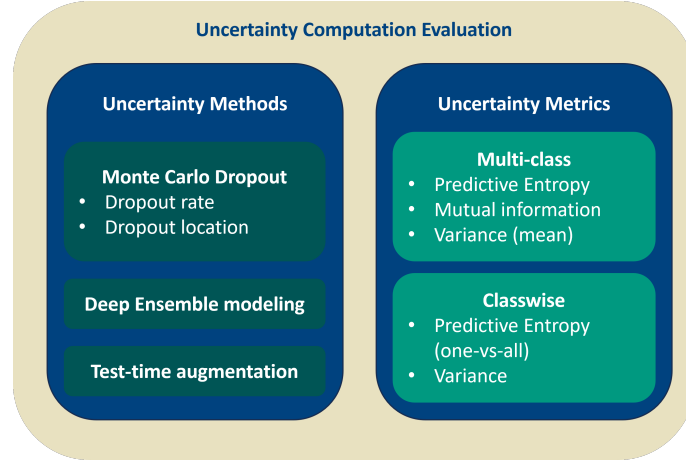


Figure 5.1: Overview of uncertainty computation methods and metrics considered in the experiments

5.3.2 Experimental Setup

The trained 3D full resolution nnU-Net [43] architecture for HNC OARs from Chapter 4 showed high performance based on geometric and dosimetric metrics. In that Chapter, we trained an ensemble of five models on 735 patients with 5-fold cross-validation and tested on a subset of 85 patients, resulting in surface DSC scores for the 19 OARs between 0.76-0.94. In this chapter, uncertainty computation methods and metrics were implemented on the fully trained network and evaluated on 10 patients from the test set. It is worth noting that the developed nnU-Net model was trained *with* data augmentation (where mirroring was disabled) but was trained *without* dropout. For more details about the network, see chapter 4.

Uncertainty Computation Methods

Three different uncertainty methods were considered for the uncertainty computation evaluation: MC-dropout, Deep Ensemble modelling and Test-time augmentation. As uncertainty metrics, predictive entropy, mutual information and variance were applied. Furthermore, a distinction between multi-class and classwise uncertainty computation was made for predictive entropy and variance. For an overview see Figure 5.1

MC-dropout

As the network was trained without dropout, dropout layers had to be subsequently inserted in the fully trained network. In the file `/nnunetv2/utilities/get_network_from_plans.py`, the variable `'dropout_op_kwargs'` was altered from 0 to the desired dropout rate for the key `'PlainConvUNet'` in the dictionary `'kwargs'`. As dropout was not implemented during training, a change in the network structure had to be made in order to insert the dropout layers. This was done in the file `nnunetv2/inference/predict_from_raw_data.py`. In the function `predict_logits_from_preprocessed_data`, names in the state dictionary (`'self.network'`) were modified by replacing all instances of `"all_modules.1."` to `"all_modules.2."` in all parameter keys (`'params.keys()'`). This adjustment was necessary as the dropout layer was inserted before the renamed layers. After renaming the parameters, the model's weights were updated. For the exact adjustments to the code, see https://github.com/joelle7/Dropout_surgery

Uncertainty computation was evaluated for dropout rates between 0 and 0.5 with intervals of 0.1, as Camarasa et al. [93] and Zhang et al. [103] found optimal performance of uncertainty to be below a dropout rate of 0.5. Dropout was applied on 'all' (MC-dropout_{all}) and the 'central' (MC-dropout_{central}) resolution stages, similarly to the method followed by Jungo et al. [79]. Where for (MC-dropout_{all}), dropout was applied on every second convolutional block in a resolution stage, except for the final stage of the decoder. For (MC-dropout_{central}), dropout was applied to the last three resolution stages of the encoder and the first three of the decoder. Dropout was implemented after the convolutional layer, before instance normalisation as mentioned by Mukhoti and Gal [67], Figure 5.2. Dropout was applied on channel-level with the nnDropout3d() class in Torch, as Zhang et al. [103] found it to perform superior compared to patch- and block-level. A total of 20 dropout output samples were obtained per patient.

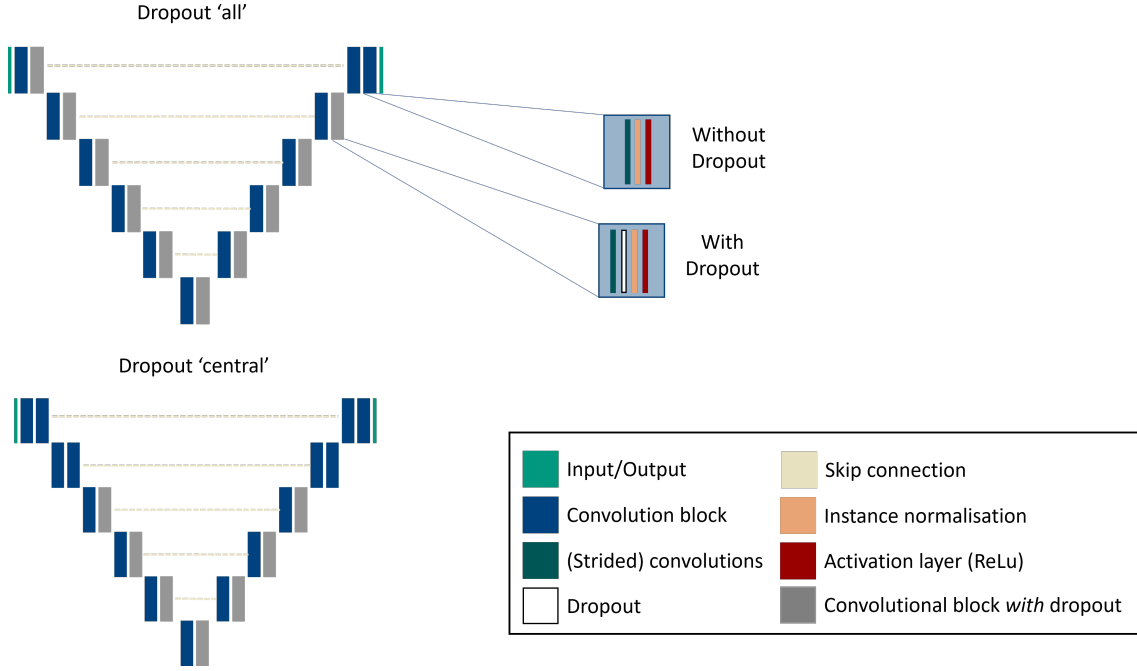


Figure 5.2: Network Architecture nnU-Net with dropout

Deep Ensemble modelling

Deep Ensemble modelling for uncertainty computation leverages multiple trained models to estimate the uncertainty of a model's output. [35] Training of the nnU-Net model in chapter 4 was performed with 5-fold cross-validation. The cross-validation models showed identical performance. The five models were used in an ensemble, similar to the method followed by Bhat et al. [98]. This resulted in 5 output samples per patient where variability is enforced, as each model was trained on an alternating fold of the training dataset.

Test-time augmentation

The same random data augmentation transformations used during training were also applied during inference to perform Test-time augmentation. [36] Augmentation probabilities and ranges are reported in the supplementary material of the nnU-Net paper by Isensee et al. [43] Data augmentations that were included are: rotation, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution and gamma augmentation. Random augmentations were applied a total of 10 times for the same patient to obtain a set of 10 output samples, similarly as was performed by Bhat et al. [98]

Uncertainty metrics

The uncertainty metrics predictive entropy, variance and mutual information were computed in Python (version 3.9.6) using NumPy (version 1.26.2) directly on the softmax output files of the DL model to compute multi-class uncertainty maps. The first two uncertainty metrics were also applied for classwise evaluation.

5.3.3 Autosegmentation performance and uncertainty map calibration

A single autosegmentation output per patient per uncertainty method was obtained by majority voting per voxel between all output samples. Also for each combination of dropout location and dropout rate, a single autosegmentation output was obtained. Results were compared against the baseline results of the nnU-Net model without uncertainty computation from chapter 4. An arbitrary threshold of a decline in autosegmentation performance of a maximum of 5% was set as acceptable. As the segmentation model predicts multiple labels, mean changes between all labels were considered and used as an assessor for the whole model’s performance.

Uncertainty maps were evaluated on segmentation performance using geometric measures and dosimetric measures. It is important to note that segmentation has a dosimetric impact on the development and evaluation of an RT treatment plan. In this study, an approximation of the dosimetric impact on evaluation of a treatment plan was considered, for ease of use, as no new treatment plans then have to be made. See Appendix section 9.5 for a more detailed explanation. The model was considered not performing up to par if any measure showed more than a 5% difference from the base model’s performance.

Uncertainty value calibration was assessed with a reliability diagram and the ECE-label value. The conventional ECE was compared with the ECE-label value in Appendix ?? and in section 9.10 the ability of uncertainty to localise segmentation errors was assessed.

5.4 Results

Figure 5.3 shows a visualisation of the computed uncertainty maps along with the CT-scan on which the segmentation was predicted, the ground truth segmentation, the predicted segmentation, an accuracy map for one patient. Uncertainty maps are visualised for multi-class entropy, mutual information and variance computed with MC-dropout with dropout rates varying between 0.1 and 0.5 applied on the central and all resolution layers.

5.4.1 Autosegmentation performance evaluation

Geometric performance

Results in Table 5.1 show the geometric performance of the autosegmentation model with uncertainty computation averaged over all OARs. Results per OAR can be found in Appendix ?. All geometric metrics remain stable compared to the base model when applying Deep Ensemble modelling and Test-time augmentation. For (MC-dropout_{central}), the performance of the HD drops more 5% compared to the base model when applying a dropout rate larger than 0.3. For (MC-dropout_{all}) a significant drop in performance can be found from a dropout rate of 0.3 for the DSC score, APL and surface DSC, however, for the HD a performance degradation of more than 5% is observed already at a dropout rate of 0.1.

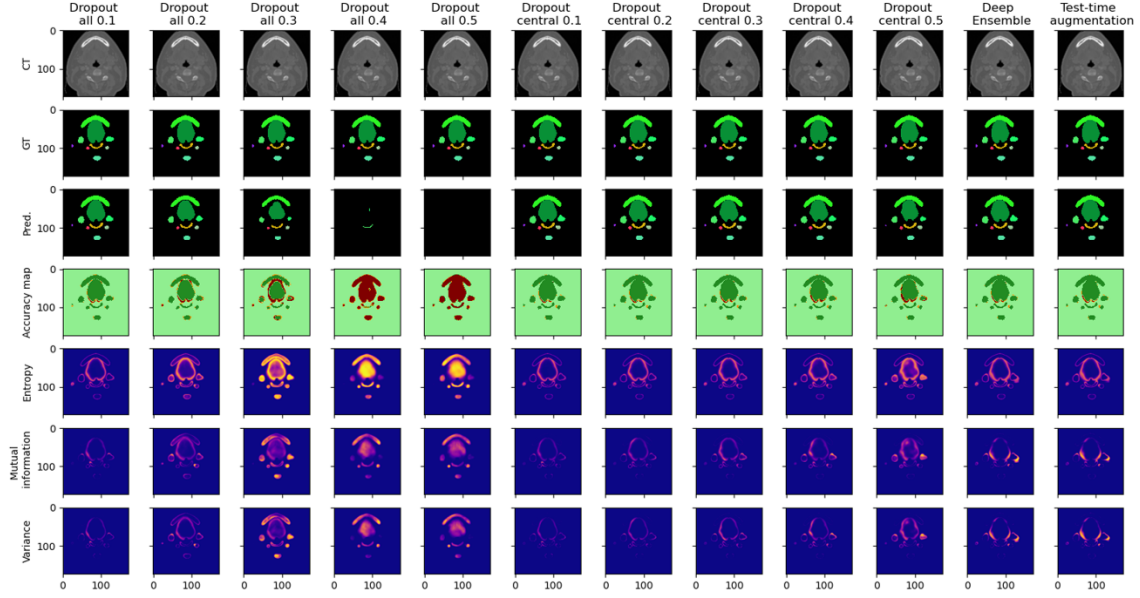


Figure 5.3: Uncertainty visualisation. Multiclass metrics. Patient 759 slice 100. Accuracy maps show TP values in forest green, TN values in light green, FN values in dark red and FP values in light red. Uncertainty maps show confidence values ranging from dark blue to bright yellow with increasing certainty. GT = ground truth, Pred = predicted autosegmentation, Dropout = Monte-Carlo dropout

Table 5.1: Autosegmentation geometric performance evaluation of nnU-Net with added uncertainty computation

Uncertainty method	Dropout location	Dropout rate	Volume DSC	HD [mm]	HD95 [mm]	APL	Surface DSC
			mean $\pm$ std	mean $\pm$ std	mean $\pm$ std		mean $\pm$ std
None / Base	-	-	0.86 $\pm$ 0.06	5.51 $\pm$ 0.41	1.62 $\pm$ 0.08	1379.13 $\pm$ 97.90	0.85 $\pm$ 0.01
Deep Ensemble	-	-	0.87$\pm$0.08	5.35$\pm$0.37	1.59$\pm$1.05	1375.95$\pm$118.14	0.86$\pm$0.02
TTA	-	-	0.87$\pm$0.01	5.48$\pm$0.39	1.59$\pm$0.10	1376.64$\pm$117.09	0.86$\pm$0.02
MC Dropout	All	0.1	0.86$\pm$0.01	5.85 $\pm$ 0.44	1.68$\pm$0.08	1312.97$\pm$96.33	0.85$\pm$0.01
MC Dropout	All	0.2	0.83$\pm$0.01	6.48 $\pm$ 0.43	1.99 $\pm$ 0.10	1331.86$\pm$101.94	0.81$\pm$0.01
MC Dropout	All	0.3	0.70 $\pm$ 0.01	9.33 $\pm$ 0.58	3.80 $\pm$ 0.25	1726.56 $\pm$ 149.94	0.60 $\pm$ 0.02
MC Dropout	All	0.4	0.13 $\pm$ 0.01	26.94 $\pm$ 1.79	21.22 $\pm$ 1.62	1525.76 $\pm$ 358.73	0.08 $\pm$ 0.01
MC Dropout	All	0.5	NaN	NaN	NaN	NaN	NaN
MC Dropout	Central	0.1	0.86$\pm$0.01	5.55$\pm$0.41	1.62$\pm$0.08	1364.94$\pm$97.19	0.85$\pm$0.01
MC Dropout	Central	0.2	0.86$\pm$0.01	5.59$\pm$0.41	1.63$\pm$0.08	1350.74$\pm$96.52	0.85$\pm$0.01
MC Dropout	Central	0.3	0.86$\pm$0.01	5.71$\pm$0.42	1.66$\pm$0.08	1334.24$\pm$96.08	0.85$\pm$0.01
MC Dropout	Central	0.4	0.86$\pm$0.01	5.92 $\pm$ 0.43	1.73 $\pm$ 0.08	1323.63$\pm$96.60	0.85$\pm$0.01
MC Dropout	Central	0.5	0.85$\pm$0.01	6.27 $\pm$ 0.43	1.91 $\pm$ 0.10	1324.76$\pm$99.67	0.83$\pm$0.01

Values in bold indicate results where the performance is no worse than 5% compared to the base model. Results with performance improvements exceeding 5% over the base model are also included. NaN occurs when no autosegmentation is obtained as output of the network anymore.

MC = Monte Carlo, TTA = Test Time Augmentation, DSC = Dice-Sørensen coefficient, HD = Hausdorff Distance, APL = Added Path length

Table 5.2: Autosegmentation dosimetric performance evaluation of networks with added uncertainty computation

Uncertainty method	Dropout location	Dropout rate	Avg. ABS Mean dose diff. ABS(CC - DLC) [Gy]	Avg. ABS Max dose diff. ABS(CC- DLC) [Gy]
			mean $\pm$ std	mean $\pm$ std
None/base		-	0.62$\pm$0.06	0.52$\pm$0.08
Deep Ensemble	-	-	0.59$\pm$0.75	0.44$\pm$0.09
TTA	-	-	0.58$\pm$0.07	0.44$\pm$0.09
MC Dropout	All	0.1	0.68 $\pm$ 0.08	0.51$\pm$0.09
MC Dropout	All	0.2	0.83 $\pm$ 0.10	0.61 $\pm$ 0.11
MC Dropout	All	0.3	1.43 $\pm$ 0.19	1.19 $\pm$ 0.16
MC Dropout	All	0.4	5.11 $\pm$ 0.79	4.85 $\pm$ 0.71
MC Dropout	All	0.5	NaN	NaN
MC Dropout	Central	0.1	0.64$\pm$0.08	0.49$\pm$0.09
MC Dropout	Central	0.2	0.65$\pm$0.08	0.50$\pm$0.09
MC Dropout	Central	0.3	0.68 $\pm$ 0.08	0.50$\pm$0.09
MC Dropout	Central	0.4	0.72 $\pm$ 0.08	0.51$\pm$0.09
MC Dropout	Central	0.5	0.82 $\pm$ 0.09	0.56 $\pm$ 0.09

Average dose differences are computed over all OARs for all test patients. Values in bold indicate results where the performance does not exceed a 5% reduction in performance compared to the base model. Results with performance improvements exceeding 5% over the base model are also included. NaN occurs when no autosegmentation is obtained as output of the network anymore.

MC = Monte Carlo, TTA = Test Time Augmentation, CC = Clinical Contour, DLC = Deep Learning Contour, Gy = Gray, Avg. = average, ABS = absolute, std = standard deviation.

Dosimetric performance

The dosimetric evaluation between the base model and models with uncertainty computations can be found in Table 5.2. Values show the absolute mean and max dose differences for all OARs averaged over all patients. Individual results per OAR can be found in Appendix ???. Dosimetric evaluation remains stable compared to the base model when applying Deep Ensemble modelling and Test-time augmentation. For (MC-dropout_{central}) an absolute mean dose difference larger than 5% is observed when applying a dropout rate larger than 0.3. For (MC-dropout_{all}) a performance degradation of more than 5% is already observed at a dropout rate of 0.1 for the absolute mean dose difference.

5.4.2 Uncertainty map calibration evaluation

Uncertainty calibration is presented separately for multi-class and classwise metrics.

Multi-class

Figure 5.4 shows the reliability diagrams per uncertainty method and metric on dataset level with the average ECE-label value between the test patients. Results for MC-dropout_{all} and for MC-dropout_{central} for a dropout rate above 0.3 are omitted, as the geometric and dosimetric performance analysis have shown that for these uncertainty computations, model performance degrades with more than 5% for one or more metrics. Deep Ensemble modelling, Test-time augmentation and (MC-dropout_{central} with a dropout rate of 0.1 and 0.2 show good reliability of uncertainty computation when using entropy, with ECE-label values of 0.06 ± 0.02 for all uncertainty methods. The other uncertainty metrics, mutual information and variance, show inferior reliability for all uncertainty methods, with ECE-label values between 0.13-0.15. The respective reliability diagrams show overconfidence for higher confidence values and no clear pattern for lower confidence values.

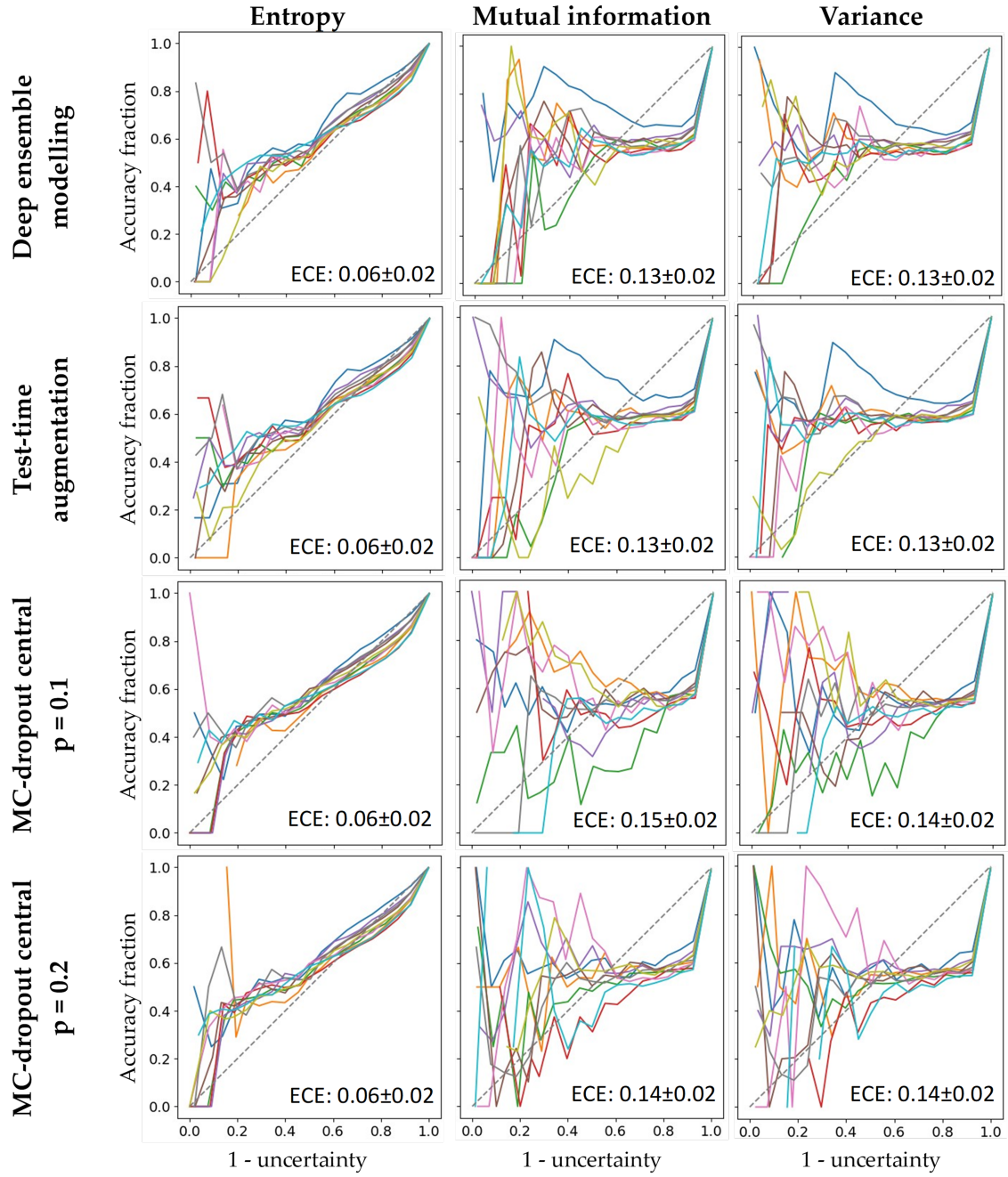


Figure 5.4: Reliability diagrams multi-class uncertainty metrics. Rows indicate uncertainty method (Deep Ensemble modelling, Test-time augmentation and MC-dropout) and columns indicate uncertainty metric (entropy, mutual information, variance). Analysis is on dataset level (10 patients), each line in a subplot represents all voxels from a single patient uncertainty map. 1 - uncertainty = confidence

Classwise

Table 5.3 presents the reliability analysis of classwise uncertainty computations for Deep Ensemble modelling, Test-time augmentation and (MC-dropout_{central} with a dropout rate of 0.1 and 0.2. Results for MC-dropout_{all} and MC-dropout_{central} for a dropout rate above 0.3 are omitted, similarly as in the multi-class analysis, due to inferior geometric and dosimetric model performance. Classwise entropy applied on Deep Ensemble modelling shows superior uncertainty computation reliability with an ECE-label of 0.06 ± 0.09 . For classwise variance, the ECE-label value is 0.09 ± 0.11 for Deep Ensemble modelling. Test-time augmentation and MC-dropout resulted in ECE-label values between 0.12-0.20. Figure 5.5 shows the classwise reliability diagrams per OAR for classwise entropy applied on Deep Ensemble modelling on dataset level. These graphs show good reliability for confidence values above 0.5, but underconfidence for values lower than 0.5. Furthermore, there is an evident difference in calibration between OARs. The uncertainty computations show poor calibration with an ECE-label value of 0.12 for the Arytenoids (left and right) and cervical oesophagus. Other evaluated OARs show better reliability of the uncertainty values, with ECE-label values ranging between 0.03-0.08.

Table 5.3: Reliability of uncertainty computations per uncertainty method and classwise uncertainty metric expressed in ECE-label value for all OARs in the segmentation model.

Organ at risk	Classwise entropy ECE-label value ↓				Classwise variance ECE-label value ↓			
	DE	TTA	MC-dropout central		DE	TTA	MC-dropout central	
			p = 0.1	p = 0.2			p = 0.1	p = 0.2
Background	0.01	0.00	0.00	0.00	0.01	0.01	0.01	0.01
A. carotid left	0.05	0.10	0.10	0.10	0.08	0.20	0.18	0.17
A. carotid right	0.05	0.11	0.11	0.11	0.08	0.20	0.18	0.18
Arytenoid left	0.12	0.25	0.27	0.27	0.14	0.32	0.31	0.31
Arytenoid right	0.12	0.24	0.26	0.25	0.13	0.30	0.29	0.29
B. mucosa left	0.07	0.15	0.15	0.14	0.11	0.26	0.24	0.23
B. mucosa right	0.08	0.15	0.15	0.15	0.11	0.26	0.23	0.23
Cricoid	0.08	0.16	0.16	0.16	0.10	0.23	0.23	0.22
Cerv. oesophagus	0.11	0.21	0.24	0.24	0.12	0.29	0.29	0.29
Glottic area	0.06	0.12	0.12	0.12	0.09	0.21	0.19	0.20
Mandible	0.03	0.05	0.05	0.05	0.03	0.07	0.06	0.07
Oral cavity	0.03	0.05	0.04	0.05	0.04	0.10	0.09	0.09
PCM	0.06	0.13	0.12	0.13	0.11	0.13	0.25	0.25
Parotid left	0.04	0.09	0.09	0.09	0.14	0.21	0.19	0.19
Parotid right	0.05	0.09	0.09	0.09	0.13	0.22	0.20	0.20
Spinal cord	0.03	0.07	0.05	0.05	0.11	0.12	0.09	0.09
Submand. left	0.06	0.11	0.11	0.11	0.11	0.19	0.18	0.17
Submand. right	0.06	0.12	0.12	0.12	0.10	0.21	0.20	0.19
Supra. larynx	0.05	0.10	0.08	0.08	0.12	0.18	0.17	0.17
Thyroid	0.05	0.10	0.09	0.09	0.09	0.19	0.18	0.18
Average of all OARs	0.06±0.09	0.12±0.06	0.12±0.10	0.12±0.09	0.09±0.11	0.20±0.11	0.19±0.11	0.19±0.10

The uncertainty method and metric combination with the highest reliability for the uncertainty maps per OAR is indicated in bold, based on the lowest ECE-label value. DE = Deep Ensemble modelling, MC = Monte Carlo, TTA = Test Time Augmentation, B. = buccal, Cerv. = cervical, PCM = pharyngeal constrictor muscle, Submand = submandibular gland, Supra = supraglottic.

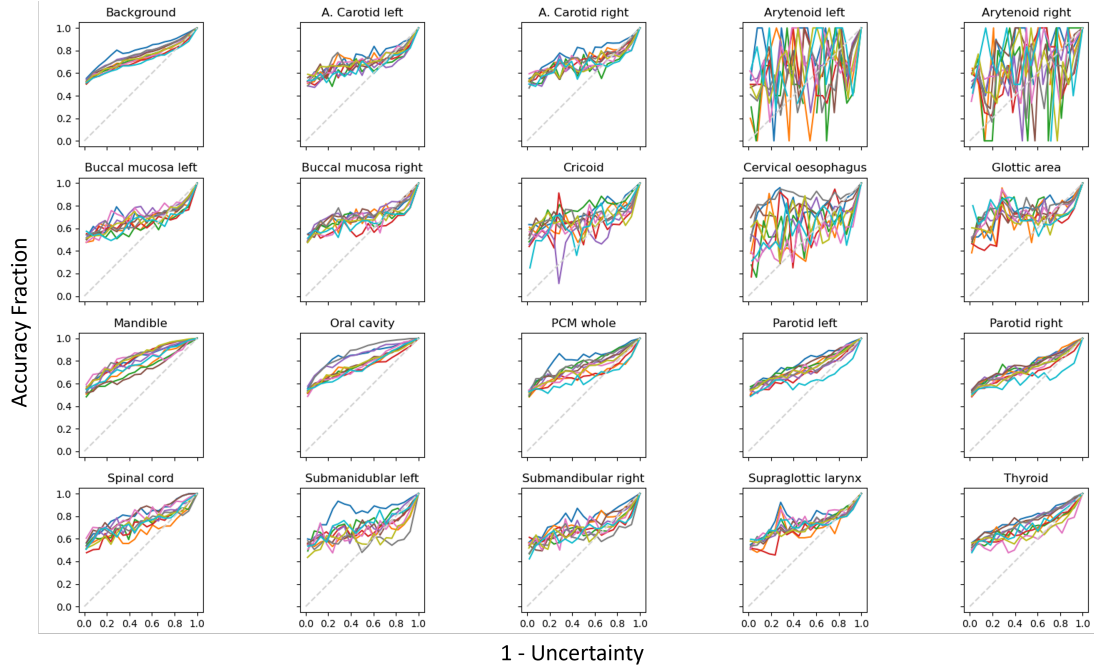


Figure 5.5: Reliability diagrams classwise uncertainty metrics per OAR for Deep Ensemble modelling with classwise entropy as uncertainty metric. The analysis is on dataset level (10 patients), each line in a subplot represents all voxels from a single patient uncertainty map. $1 - \text{uncertainty} = \text{confidence}$

5.5 Discussion

Uncertainty computation could assist autosegmentation acceptance in radiotherapy and realise the potential time gain and performance consistency possible with DL. Quality of an uncertainty map is essential for QA purposes, however, no consensus exists on the uncertainty method and uncertainty metrics that provide the most reliable uncertainty maps. An analysis of the quality of uncertainty computation using different uncertainty methods *and* metrics applied to OAR autosegmentation in radiotherapy was lacking. This study has provided an evaluation of the three most popular uncertainty estimation methods (MC-dropout, Deep Ensemble modelling, and Test-time augmentation) in combination with the three most popular uncertainty metrics (predictive entropy, mutual information and variance). Uncertainty computation showed stable implementation for Deep Ensemble modelling, Test-time augmentation and MC-dropout_{all} for a dropout rate up to 0.2. Dropout rates higher than 0.3 or when using MC-dropout_{all} resulted in considerable model performance reduction. Entropy as an uncertainty metric performed superior with similar uncertainty value calibration on all uncertainty methods. Variance and mutual information showed poor calibration. This indicates that the uncertainty method is a negligible hyperparameter compared to the uncertainty metric.

The geometric and dosimetric analysis showed no degradation in performance of all considered geometric and dosimetric metrics when implementing uncertainty computation with Deep Ensemble modelling, Test-time augmentation and MC-dropout_{central} with a dropout rate of 0.1 and 0.2.

Geometric quality of the autosegmentation performance was only significantly influenced for MC-dropout_{all} when using a dropout rate of 0.2 or higher for the volumetric DSC score, APL and surface DSC. HD showed already degraded geometric performance when using a dropout rate of 0.1 on all resolutions. For MC-dropout_{central}, a drop in performance

of more than 5% was observed above a dropout rate of 0.3 only for HD. A considerable decrease in autosegmentation performance indicates that careful consideration of this hyperparameter tuning is advised when implementing uncertainty computation, as was also mentioned by Gal et al. [94]. It shows that an essential part of the knowledge of the model is dropped. When computing uncertainty with Deep Ensemble modelling and Test-time augmentation, an improvement in geometric performance for all geometric measures was observed. This is expected behaviour, as these methods were originally developed to improve model performance. [35, 36, 37]

Absolute mean and max dose differences remained within 1.0 Gy for most uncertainty methods, except when applying a dropout rate of 0.3 on MC-dropout_{central} or for MC-dropout_{all} on all dropout rates. For the latter cases, the average mean dose difference was more than 5% than observed with the base model, thus indicating a significant influence of the uncertainty computation method on dosimetric autosegmentation quality. The geometric and dosimetric results show that reliable implementation of uncertainty computation is only realised with Deep Ensemble modelling, Test-time augmentation and MC-dropout_{central} up until a dropout rate of 0.2. For further analyses, uncertainty computation can be limited to these methods.

The usability of the uncertainty values was evaluated for the uncertainty methods with stable geometric and dosimetric performance (i.e. Deep Ensemble modelling, Test-time augmentation and MC-dropout on central resolution stages with a dropout rate of 0.1 and 0.2). The reliability analysis included reliability diagrams and the ECE-value. From the reliability diagrams, ECE-label values and visual evaluation of the uncertainty maps in Figure 5.3, it is clear that the different uncertainty metrics lead to distinctly different uncertainty maps and reliability. Superior calibration was observed when using predictive entropy (ECE-label value of 0.06) compared to variance (0.13-0.15) or mutual information (0.13-14). For all uncertainty methods ECE-label values were highly comparable with a difference of at most 0.02 when using the same uncertainty metric. This highlights that the uncertainty method is a negligible hyperparameter compared to the uncertainty metric, where differences of 0.07 to 0.09 were observed for the same uncertainty methods.

Entropy is, in theory, more appropriate for non-unimodal distributions, compared to variance. This is the case in multi-label classification, so this could explain the superiority of entropy. Mutual information is theoretically specifically suited for aleatoric uncertainty (i.e. data uncertainty). Since not all errors in the autosegmentation are due to aleatoric uncertainty, but also due to epistemic uncertainty or an entangled combination, this could explain why mutual information performed inferior. Nevertheless, Test-time augmentation allegedly *also* highlights areas of aleatoric uncertainty, but *did* show identical reliability (with predictive entropy) to MC-dropout and Deep Ensemble modelling.

The superior performance of entropy, as observed in this study, aligns with the findings of Camarasa et al. [93]. They compared various classwise and multi-class uncertainty metrics solely on MC-dropout and found superior performance for predictive entropy compared to variance, but also for mutual information for multi-class uncertainty maps. They found entropy to perform superior for class-wise uncertainty, consistent with this study's results. Their observation that multi-class mutual information performs similarly to multi-class entropy might be due to their use of the AUC-PR score after thresholding the uncertainty map instead of the ECE(-label) value, which was performed in this research.

In contrast, a previous study by Sahlsten et al. [81] found that variance performed superior to entropy and mutual information in accuracy detection. Nair et al. [71] performed a similar analysis, evaluating entropy, mutual information and variance on MC-dropout and observed similar performance for all metrics. A possible explanation for this is that

Sahlsten et al. and Nair et al. did not evaluate the calibration of uncertainty maps but focused on outlier and failure detection. This highlights that calibration and uncertainty usability are distinct analyses.

Wahid et al. [33] further illustrate the lack of consensus on uncertainty metrics. Their study found that 33.9% of research in autosegmentation for radiotherapy applied variance-based metrics, 27.4% used entropy, 16.1% employed probability-based metrics, and 22.6% utilized other methods. Most studies focus on applying a single uncertainty metric without evaluating calibration, even though calibration is crucial for ensuring that uncertainty values accurately reflect a model’s confidence, especially for clinical applications in QA processes. This shows that uncertainty calibration evaluation is often neglected, despite its importance for safe application of uncertainty.

It is interesting to note that with classwise uncertainty a decrease in average ECE-label value was observed for Test-time augmentation and MC-dropout compared to multi-class metrics, Table 5.3 and Figure 5.5. For Deep Ensemble modelling, the average ECE-label value for classwise entropy is identical to multi-class entropy ECE-label value (0.06). However, the standard deviation is larger. A similar observation was made by Camarasa et al. [93] who found multi-class to perform superior to class-wise uncertainty metrics. Detangling the uncertainty per OAR in a segmentation network is thus not trivial. The larger ECE-label value and standard deviation indicate a discrepancy between organs in reliability, resulting in inferior reliability of uncertainty computation overall.

For the classwise uncertainty maps, the reliability of the uncertainty values for the arytenoids and cervical oesophagus is inferior compared to the other organs. The ECE-label values are 0.12 and 0.11 compared to 0.03-0.08 for the other OARs, when considering Deep Ensemble uncertainty computation with classwise entropy. This trend is also observed for the other uncertainty methods and metrics. A possible explanation can be found for arytenoids in their size. They are small structures with relatively high fractions of falsely segmented voxels compared to bigger OARs. Nair et al. [71] similarly observed higher uncertainties and inferior performance for smaller brain lesions. They explained this by noting that a large amount of voxel-level evidence (i.e. bigger structures) eases the localisation of a region in a network compared to smaller lesions. For the cervical oesophagus, the discrepancy between errors could be drawn back to the earlier observation of inadequate cranial or caudal extension of the structure, related to a change in delineation guideline interpretation during the data accumulation period.

Another interesting observation is the y-intercept of the reliability diagrams, i.e. the accuracy fraction of the voxels with the highest uncertainty/lowest confidence. For classwise metrics, this value lies around 0.4. This means that for voxels that have been assigned an uncertainty of 95-100 %, 40 % are still accurately segmented, and uncertainty is thus under-confident. For higher confidences, calibration values are more representative of the accuracy fraction. This, however, follows the expected behaviour of uncertainty described by Mukhoti and Gal [67]. They stated that a model is expected to be correct more often when uncertainty is low, but may or may not be accurate if the uncertainty is high, also see the Methods section section 5.3.1. This indicates that uncertainty computation prioritises the correctness of certainty statements against uncertainty statements.

While MC-dropout is typically applied during inference on models trained *with* dropout [34], we observed that our model, trained *without* dropout, maintained stable performance and well-calibrated uncertainty maps, comparable to other uncertainty methods. This stability may be due to the robustness of the trained nnU-Net, which does not seem to rely on critical regions that would otherwise compromise the model’s functionality. As shown in chapter 4 Figure 4.3, the loss curves and validation DSC indicate stable model training.

These results suggest that inserting MC-dropout into a network trained without dropout is indeed feasible.

Limitations of this study include the limited set of 10 patients analysed. However, no clear performance imbalance was observed between subjects. Only for classwise entropy a relatively large overconfidence was observed for one patient (light blue line) for the left and right parotid and another patient (dark blue line) observed a relatively high under-confidence for the left submandibular gland. When further evaluating the imaging data of these patients, it becomes clear that these were visually challenging cases due to metal artefacts in the oral cavity. Another limitation is that only three uncertainty methods and three uncertainty metrics were evaluated and emerging uncertainty methods were neglected, such as deterministic uncertainty [104]. MC-dropout, Deep Ensemble modelling and Test-time augmentation were selected based on existing prevalence in autosegmentation uncertainty computation literature.[33] However for future research it might be interesting to evaluate new emerging methods for uncertainty computation such as deep deterministic uncertainty [104].

The Jaccard index was used to define accuracy to compute the calibration error, to minimise the effect of the large volume of TN values from the background. However, other metrics for imbalanced datasets, such as the F1 score, could also be considered. Furthermore, this study only evaluated the influence of uncertainty computation on model performance and the calibration of uncertainty values. No evaluation of the usefulness of uncertainty values was performed, for example for autosegmentation failure detection. This is however important to analyse as this concerns a possible purpose for uncertainty in an automated RT workflow for QA. A topic of discussion in uncertainty for analysing the generalisability of DL segmentation in RT, is the theoretical value of uncertainty. No consensus exists on what each type of uncertainty visualises and if theoretical ideas of uncertainty hold in practical observations. A deeper investigation into what these different uncertainty metrics show and if it aligns with their theoretical differences, is an area which the next section will explore.

5.6 Conclusion

This study implemented the three most popular uncertainty computation methods (Deep Ensemble modelling, Test-time augmentation and MC-dropout) and uncertainty metrics (entropy, mutual information and variance) for RT autosegmentation on multi-class and classwise levels. We analysed the influence of uncertainty computation on model performance with a set of geometric and dosimetric measures and calibration of uncertainty maps using reliability diagrams and an adjusted ECE value (ECE-label). MC-dropout_{central} with a dropout rate larger than 0.2 and MC-dropout_{all} in general lead to a decrease in geometric and/or dosimetric performance larger than 5%. Lower dropout rates, Deep Ensemble modelling and Test-time augmentation demonstrated stable performance.

We showed that the uncertainty method is a negligible parameter compared to the uncertainty metric, as multi-class entropy resulted in superior calibrated uncertainty values compared to mutual information and variance for all uncertainty methods. Classwise entropy demonstrated similarly high calibration of uncertainty values as multi-class entropy on Deep Ensemble modelling, but did show a degradation in calibration when applied to other uncertainty methods. Classwise variance showed inferior performance on all uncertainty methods.

Chapter 6

Uncertainty for Out-of-Distribution Detection

6.1 Introduction

Although autosegmentation in RT has reached the accuracy of manual segmentation, [57] time-gain and reproducible quality by minimising inter- and intra-observer variability have been minimal, as still many adjustments are being made. [46]. Inadequate implementation possibly exists due to a lack of trust in segmentation model output. This issue is compounded by the complicated environment of RT where segmentation inaccuracies have real-life consequences considering tumour control and RT-related toxicity. [7]

Uncertainty computation could improve confidence in an autosegmentation model and aid integration in the clinical workflow through QA, by stating how confident a model is about its about. Furthermore, a prospect in RT is online adaptive treatment, where a high-paced composition of a treatment plan is necessary for real-time modifications to a patient’s treatment plan. Uncertainty thus could play an indispensable role in DL segmentation model implementation in the clinic, by communicating when it ‘does not know’. [33, 105]

However, what it means when a model ‘does not know’ is an area of discussion in computer vision AI models. Various uncertainty methods and metrics have been proposed to represent different types of uncertainty. Conventionally there are two types of uncertainty: epistemic (i.e. model) and aleatoric (i.e. data) uncertainty. [34] Kendall and Gal [38] describe epistemic uncertainty to reflect out-of-distribution examples that fall outside of training data (i.e. a lack of information in the model for accurate prediction). They refer to aleatoric uncertainty as inherent stochasticity of the environment (i.e. in the data) that cannot be explained away by expanding the training data set, only if the stochasticity is reduced by for example a reduction in sensor noise.

However, in RT applications, there is no consensus on uncertainty computation disentanglement and the translation of theoretical uncertainty representation to a practical level, as described in Section 3.3.3. While some authors describe aleatoric uncertainty as a representation of inter-observer variability caused by the stochastic component in the ground truth delineations or noise in the (input) CT-scans [7, 39], others refer to aleatoric uncertainty as a method to detect OOD samples [40, 41]. Epistemic uncertainty reflects the limitation of an artificially created model to represent a real-world process [39]. According to some authors, this represents the limited data set a model is trained on and is uncertain when a sample is outside the training data set distribution. [7, 39]. While others argue

that it represents which regions have been wrongly segmented in general. [42]

Uncertainty disentanglement and translation from theoretical understanding to practical application is thus not trivial. However, Charpentier et al. [74] argue, that a division of uncertainty into aleatoric and epistemic uncertainty is key for reliable deployment of DL models in real-world application. While otherwise, AI generalisability assessment and OOD detection are unfeasible. Wahid et al. [33] state that specifically epistemic uncertainty estimation is important for RT model development, as models trained in medicine are often trained on considerably small data sets compared to other computer vision models outside of medicine.

In the clinical application of AI in RT, the priority might not be to distinguish between epistemic and aleatoric uncertainty per se, as this is a more theoretical problem. However, it is interesting to investigate the practical value supposedly obtained by disentangling uncertainty. It could be of great value to visualise if an error lies within the inter-observer variation or arises due to imperfection of the segmentation model (e.g. insufficient amount of similar training data). While keeping in mind that the primary goal of uncertainty in the automated RT workflow, is to help the clinician reliably and consistently correct autosegmentation results.

Previous studies have investigated the relationship between uncertainty and inter-observer variation [76, 106], as well as the effectiveness of uncertainty in detecting segmentation failures [33]. However, less is known about the relation between uncertainty and OOD.

Benchmarks showing the ability of uncertainty estimation for OOD detection are often limited to heavily controlled environments with toy data. [32, 37, 35] Opposed to these experiments, medical image data is characterised by more nuanced real-world differences than the domain shifts typically evaluated. Linmans et al. [105] have applied uncertainty estimation for OOD detection in digital pathology and Lakara et al. [107] applied it to medical generative models for MRI scans. Both found uncertainty estimates to be able to discriminate between ID and OOD data. However, similar research investigating the usability of uncertainty in OOD detection has not been applied in the RT autosegmentation domain to the best of our knowledge. As data distribution in RT is different from digital pathology or MRI generative models, it is interesting to investigate the ability of uncertainty to perform OOD detection in RT as well. Especially as uncertainty could fulfil an important role in QA for autosegmentation failure detection and adaptive RT.

This study investigates the ability of uncertainty to perform OOD detection on an autosegmentation model for HNC OARs with real-world OOD data. We aim to understand the influence of data distribution on uncertainty computation and validate expected uncertainty behaviour. For this, we formulate a set of RT-specific qualifications on CT scans that define OOD data (metal artefacts, bolus and bitelock). We evaluate uncertainty computation for two types of uncertainty computation methods (MC-dropout and Deep Ensemble modelling and compare their capability for OOD detection.

6.2 Background

The classic conception is that epistemic uncertainty reflects the gap in a model’s knowledge due to the restricted or incomplete data available during training. [34] In computer vision, this is often translated to a lack of experience with similar visual data during training (e.g. MNIST vs Fashion MNIST dataset [108]) or because the image is visually challenging or ambiguous (e.g. shadows [38] or artificial corruptions as in the ImageNet-C/P dataset). These concepts are typically translated as far and shifted OOD data respectively. [109] In benchmark OOD datasets for segmentation, such as the Segment-Me-If-You-Can dataset,

far OOD is captured as artificial anomalies in the image and shifted OOD is represented by challenging conditions such as nighttime or extreme weather.

OOD detection on real-world medical data has been limited. Linmans et al. [105] developed a model for lymph node classification on pathological data and a model for prostate biopsy classification on pathological data. They evaluated the ability of uncertainty to detect unseen-before types of lymphoma and classify prostate tissue in the presence of unseen-before colorectal tissue in the respective models. Lakara et al. [107] developed a generative model for image translation (T1-weighted brain MRI scans to T2-weighted brain MRI scans) and evaluated the ability of uncertainty to detect unseen scan types (brain CT scans and the UTK face dataset).

However, concepts for OOD data have not been developed for real-world examples in the radiotherapy domain. We now formulate three cases of OOD data for the task of OAR in the head-and-neck region on CT images. A bitelock can be present in the oral cavity for oropharyngeal cancer patients to minimise dose deposition on healthy tissue. When a model is only trained on patients without a bitelock, patients with a bitelock can be seen as anomalies in the images and thus as *far OOD*. Similarly, a bolus can be put on the skin to overcome the skin-sparing effect, providing another definition of *far OOD* if the model has not seen this data before. Secondly, patients can have dental implants or metal fillings that cause metal artefacts in the CT images, thus hampering clear identification of surrounding structures. When a model is only trained on patients without metal artefacts, patients with metal artefacts can be seen as *shifted OOD*, as the model has experienced similar structures before, however, the CT scan now is visually challenging.

6.3 Problem setup

This study evaluates the ability of different uncertainty estimation methods to perform OOD detection. For this, the model performance and uncertainty computation results are compared between both ID and OOD data for three definitions of OOD data (metal artefacts, bolus and bitelock). A separate model is developed for each type of OOD data by withholding the specified subset of OOD data.

Dataset

Three arbitrarily defined subsets of OOD data were created within the data distribution for HNC patients receiving RT as used in Chapter 4, consisting of 812 patients. This dataset was combined with an additional dataset with similar characteristics of 268 patients treated between 2007 and 2018, leading to a total of 1080 patients. Patients with metal artefacts, patients with a bolus and patients with a bitelock were defined as OOD subsets, as shown in Figure 6.1. This led to a total of 470 patients with metal artefacts, 251 patients with a bolus and 19 patients with a bitelock.

A new version of the nnU-Net segmentation model for 19 HNC OARs was trained with only ID data for all three subset definitions, $\hat{y} = F_{OOD}(x, W_{in})$, with W_{in} the model weights and F_{OOD} the OOD model (i.e.. the no-metal-model, no-bitelock-model and no-bolus-model). Let x_{in} define an ID input CT-scan and $\hat{y}_{in}$ the corresponding in-distribution segmentation prediction and y_{in} the ground truth segmentation. Let x_{out} , $\hat{y}_{out}$ and y_{out} define the same for a subset of patients that were held out during training, the OOD patients. We evaluated the ability of uncertainty estimates from the OOD model (F_{OOD}) to discriminate between ID and OOD data using a set of 10 ID and 10 OOD test patients and compared it with the baseline model.

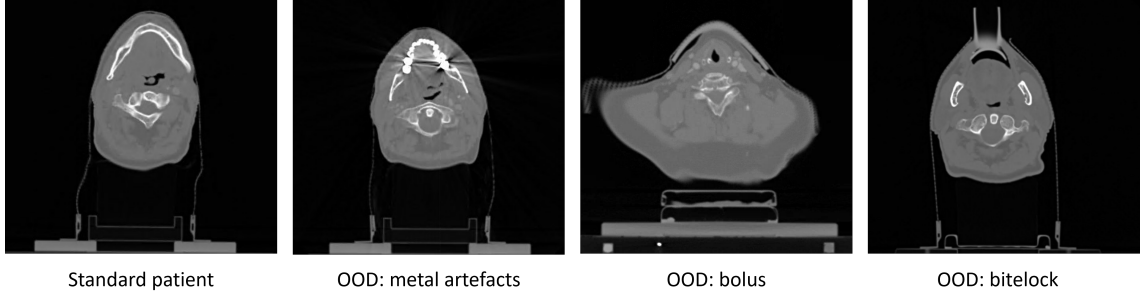


Figure 6.1: Definition of out-of-distribution patient subsets: metal artefacts, bolus and bitelock

Model

All three OOD nnU-Net models were trained according to the method described in chapter 4 using cross-validation with an in-distribution dataset $D_{in}\{x_{in}, y_{in}\}$. Due to limited data availability of ID patients, five custom cross-validation splits of 589 training patients and 12 validation patients were defined for all models. The models were trained without mirror-augmentation to account for bilateral OARs for 1000 epochs to obtain a 3D full-resolution nnU-Net. Training was performed on the Hábrók high-performance computer cluster from the Centre for Information Technology of the University of Groningen, with one Nvidia A100 GPU accelerator card with 512GB memory (Python 3.9.6, Pytorch 2.1.2 with CUDA 12.1.1). For more information, see chapter 4 and Appendix section 9.3.

6.4 Methods

We evaluate the capacity of uncertainty estimations to perform OOD detection on methods and metrics that have shown reliable implementation in the previous section on an nnU-Net model [43] for HNC OARs developed in the second part of this thesis in Chapter 5. This section describes the implemented uncertainty computation methods and metrics as well as the experiments that evaluate the ability of uncertainty to detect OOD data.

6.4.1 Uncertainty estimation methods

Uncertainty computation methods were implemented on the baseline model and all three OOD models (i.e. the no-metal model, no-bitelock model and no-bolus model) following the procedures detailed in Chapter 5. Deep Ensemble modelling and MC-dropout on the central resolution stages with a dropout rate of 0.2 were selected as uncertainty methods, as these methods showed stable model performance and reliable uncertainty value calibration. Unfortunately, due to the limited time available, Test-time augmentation was not implemented.

6.4.2 Uncertainty estimation metric

Entropy was the only uncertainty metric that showed reliable uncertainty values in chapter 5 compared to variance and mutual information. Therefore, to assess the confidence for each model output, only entropy is implemented, according to:

$$Entropy = - \sum_{c=0}^C \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right) \log \left(\frac{1}{T} \sum_{t=1}^T p(\hat{y} = c | x_i^*, W_t) \right) \quad (6.1)$$

With $\hat{y}$ the predicted output class, x_i^* the test input, C the number of classes, T the number of uncertainty samples, i the i^{th} patient and W the network weights.

6.4.3 Desiderata and experiments

This research focuses on the usability of uncertainty for OOD detection. We defined a set of desiderata to reflect the expected behaviour of uncertainty estimates between ID and OOD samples. These desiderata address model performance accuracy, uncertainty reliability, and the ability to discriminate between ID and OOD data. For readability the desiderata are explained using the no-metal model as an example, however, the definitions are directly translatable for the no-bolus and no-bitelock models.

Desideratum 1.1 *A model with a complete training environment and a model with a limited training environment should show similar performance accuracy on a subset of the data that was part of the training environment in both models*

Intuitively, the baseline model and the no-metal model should have similar performance on ID patients (without metal artefacts), as this subset of the data was part of the training environment in both models. In practice des. 1.1 checks whether model performance is maintained despite excluding a subset of data. Experiment 1.1 compared geometric performance on 10 ID patients between the baseline and the no-metal model based on the surface DSC score. It was expected that the patients show similar surface DSC scores between the baseline model and the no-metal-model.

Desideratum 1.2 *A model should show worse performance on a subset of the data that was not part of the training environment compared to data within the training environment*

Intuitively, performance should decline on OOD data relative to ID data. In practice, des 1.2. validates if the OOD definition is effective. Experiment 1.2 compared performance on 10 ID versus 10 OOD patients for the no-metal-model based on the surface DSC score. It was expected that the surface DSC for OOD patients is lower compared to ID patients on the no-metal-model.

If the model did not show a reduction in performance for OOD data compared to ID data, the analysis could stop here, as this would indicate that the definition of OOD used in this study was ineffective to distinguish between known and unknown domains. However, if there was a performance discrepancy, the analysis could continue. Then, firstly it was established that uncertainty computation was still reliable.

Desideratum 2.1 *Uncertainty value calibration should remain reliable for a model trained on a limited portion of the training environment, explicitly also on data outside the training distribution*

Intuitively, uncertainty values should reflect model confidence for both ID and OOD outputs. In practice, des. 2.1 ensures that the uncertainty estimates from the no-metal-model are meaningful and consistent with those from the baseline model for both ID and OOD data. Experiment 2.1 compared ECE-label values for 10 ID and 10 OOD patients both on the baseline model and no-metal-model. It was expected that both the ID and OOD patients show similar uncertainty calibration levels to the baseline model.

If uncertainty maps were well-calibrated and OOD performance was reduced in the no-metal-model, the next step could assess whether uncertainty estimates are able to effectively distinguish between ID and OOD samples. For a quantitative comparison, metrics can be obtained from the uncertainty histograms.

Desideratum 3.1 *A model with a complete training environment and a model with a*

limited training environment should show similar uncertainty values on a subset of the data that was part of the training environment in both models

Intuitively, both the baseline and the no-metal-model should show comparable uncertainty for ID data. In practice des. 3.1 validates that uncertainty computation values for the no-metal-model are calibrated with the baseline model. Experiment 3.1 compared uncertainty histograms for 10 ID patients between the baseline and no-metal model. It was expected that the patients show similar uncertainty levels between the models.

Desideratum 3.2 *A model should show higher uncertainty for data outside the training environment compared to those within*

Intuitively a model should be less confident (i.e. show more uncertain) when observing new states that were not encountered within the training distribution. In practice des. 3.2 validates that a model's confidence reflects if an encountered state was within the knowledge of the model. Experiment 3.2 analysed uncertainty histograms for 10 ID and 10 OOD patients on the no-metal model. It was expected that OOD patients exhibit higher uncertainty.

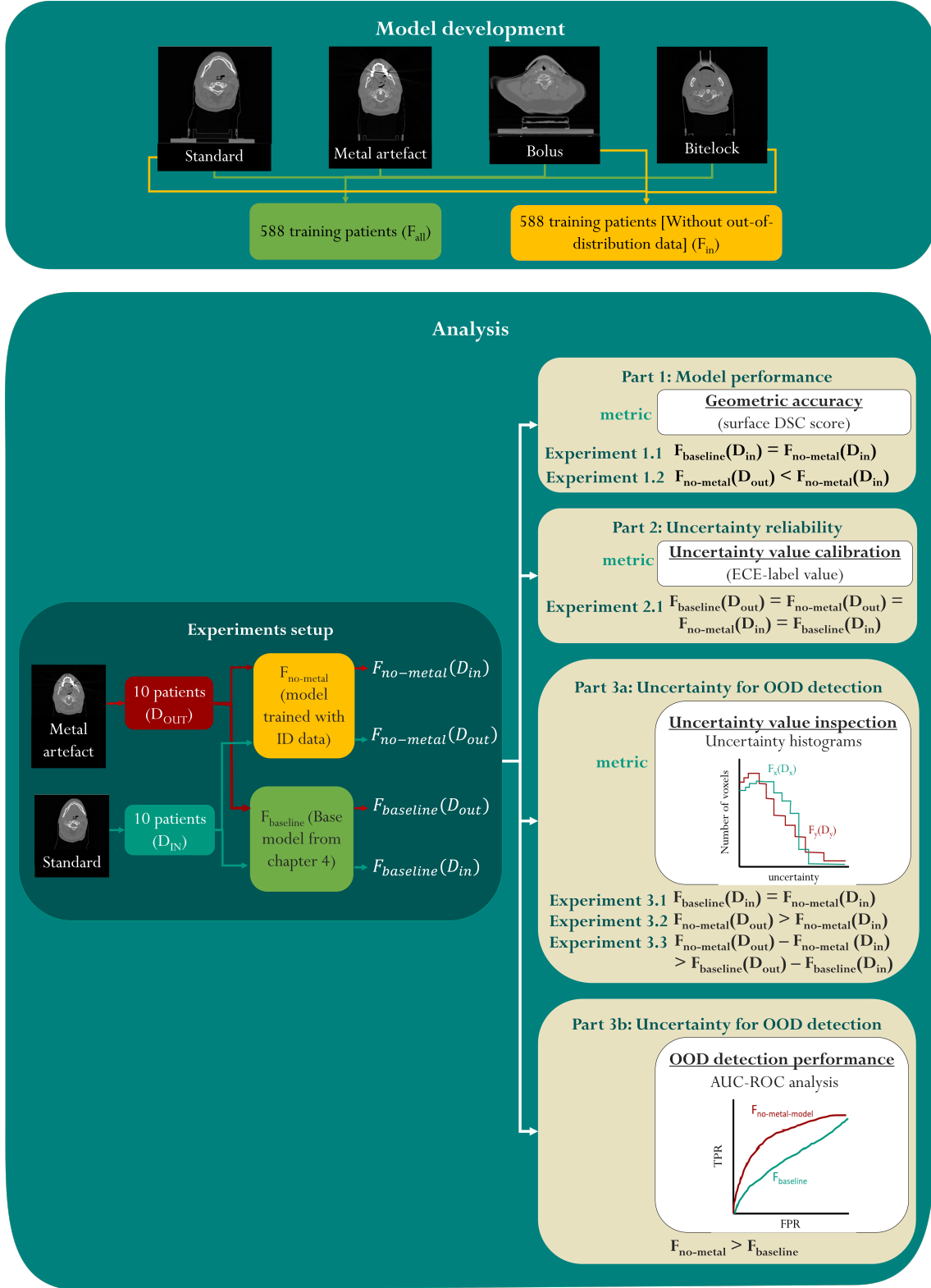
Desideratum 3.3 *A model trained on a subset of the training environment should demonstrate better discrimination between data within and outside the training distribution compared to a model trained on the complete training environment.*

Intuitively, a model trained on a subset of the training environment should exhibit greater uncertainty (i.e., be less confident) on patients outside the training distribution compared to a model trained on the entire environment. Conversely, for a model trained on the complete training environment, the confidence on this subset should be comparable to that on a standard set of patients within the training environment. In other words, a model with a complete training environment should be less effective at distinguishing between ID and OOD data.

In practical terms, Desideratum 3.3 aims to validate whether observed higher uncertainty for OOD patients is indeed model-dependent rather than data-dependent. Specifically, Experiment 3.3 compares the uncertainty histograms between 10 OOD and 10 ID patients between the no-metal model and the baseline model. It is anticipated that the no-metal model will show a greater difference in uncertainty between ID and OOD patients compared to the baseline model.

Finally, the OOD detection performance is measured by conducting an AUC-ROC analysis. In this analysis, the ability of the uncertainty histogram metrics to distinguish between OOD and ID labels for a set of 10 ID and 10 OOD patients on the no-metal-model. This can be also done for the baseline model, where a lower AUC value is expected.

For an overview of the methods, see Figure 6.2


 Figure 6.2: Method overview for analysing suitability of uncertainty for **ODD!** (**ODD!**)

D_{out} = out-of-distribution patients (not included in data distribution of model during training)

D_{in} = in-distribution (included in data distribution of model during training)

F = segmentation model, F_{all} = base segmentation model trained on full data distribution. $F_{no-metal}$ = segmentation model trained only on in-distribution patients, i.e. no patients with metal artefacts. DSC = Sørensen Dice Similarity coefficient, ECE = expected calibration error, TPR = true positive rate, FPR = false positive rate, D_{out} = out-of-distribution data, D_{in} = in-distribution data

6.5 Results

This section presents the outcomes of the OOD detection experiments formulated by the desiderata. Table 6.1 provides a summary of the surface DSC scores and ECE-label values for both ID and OOD data across all OOD models (no-metal, no-bolus, and no-bitelock models), with a comparison to the baseline model. The results evaluate Desiderata 1.1, 1.2, and 2.1. We focus on a subset of OARs, chosen for their susceptibility to artefacts due to their anatomical location. A comprehensive analysis of all OARs is available Appendix ??.

Figure 6.4 presents the uncertainty histograms for ID and OOD data on the no-metal model, no-bolus model and no-bitelock model. Various uncertainty histogram metrics were evaluated, including (thresholded) mean uncertainty, (thresholded) standard deviation and voxels above a threshold. Best performance was observed for the number of voxels with an uncertainty above a threshold $T = 0.3$ ($N_{voxels,U>0.3}$) and the mean uncertainty were computed. A full analysis can be found in Appendix Figure 9.11. Figure 6.5 reports the mean uncertainty value and the number of voxels above a threshold of 0.3 ($N_{voxels,U>0.3}$). These results evaluate Desiderata 3.1, 3.2, and 3.3. The threshold of 0.3 was chosen based an evaluation of the thresholds 0.01, 0.05, 0.1, 0.2, 0.3 and 0.5.

6.5.1 Analysis 1.1: Model performance on ID patients

Experiment 1.1 evaluated whether the OOD models matched the baseline model’s performance on ID data. Table 6.1 demonstrates that the no-metal model has stable performance with the baseline model with maximum deviations in the surface DSC scores of 0.02 for the left buccal mucosa, mandible, left and right parotid glands and left submandibular gland. Slightly larger deviations (0.03 to 0.05) were observed for the right buccal mucosa, oral cavity and right submandibular gland. The no-bolus and no-bitelock model performed consistently with the baseline model on ID data with maximum deviations ≤ 0.02 .

6.5.2 Analysis 1.2: Model performance on OOD patients

Experiment 1.2 evaluated whether performance of OOD patients was inferior to ID patients on OOD models. Table 6.1 shows that the no-metal model experienced a performance decrease of 0.03 to 0.09 in surface DSC scores for the right buccal mucosa, mandible, oral cavity, left and right parotid glands, and left submandibular gland in metal artefact patients. Other OARs showed similar performance between ID and OOD patients. The no-bolus model experienced a performance drop of 0.03 to 0.08 for the oral cavity, left parotid gland, and left and right submandibular glands for bolus patients compared to ID patients. The no-bitelock model exhibited reductions of 0.04 to 0.12 in the surface DSC score across all OARs in bitelock patients.

For the mandible, oral cavity, and right parotid gland the performance of OOD patients on the no-metal model was not only inferior to ID, but performance was also inferior to the same OOD patients on the baseline model, with deviations in surface DSC score of 0.04 to 0.06. Similarly, the no-bitelock model demonstrated a decrease of 0.03 for the left and right submandibular glands on OOD patients compared to the baseline model. Performance for other OOD OARs remained consistent between the OOD and baseline models.

6.5.3 Analysis 2.1: Uncertainty reliability

Experiment 2.1 examined the reliability of uncertainty computation for both ID and OOD data on the OOD models. Table 6.1 shows that the ECE-label values for ID data remained stable across both the baseline and OOD models, indicating well-maintained uncertainty

calibration. However, uncertainty reliability declined for bolus and bitelock OOD patients, with ECE-label values increasing to 0.10-0.18 both on the baseline and OOD models. Notably, uncertainty computation for metal artefact patients remained reliable in both the baseline and no-metal models.

Table 6.1: Model performance and uncertainty reliability analysis for ID and OOD on the baseline model against OOD models

	Metal artefacts				Bolus				Bitelock			
	Baseline model		No-metal model		Baseline model		No-bolus model		Baseline model		No-bitelock model	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD
Avg. surface DSC												
Buc. mucosa L	0.78	0.75	0.76	0.73	0.78	0.75	0.78	0.76	0.78	0.68	0.76	0.67
Buc. mucosa R	0.78	0.76	0.73	0.70	0.78	0.77	0.78	0.76	0.78	0.70	0.78	0.69
Mandible	0.99	0.97	0.98	0.93	0.99	0.98	0.99	0.98	0.99	0.95	0.99	0.95
Oral cavity	0.84	0.80	0.81	0.74	0.84	0.77	0.84	0.76	0.84	0.78	0.86	0.77
Parotid left	0.84	0.77	0.83	0.76	0.84	0.81	0.83	0.80	0.84	0.71	0.82	0.70
Parotid right	0.79	0.76	0.78	0.72	0.79	0.78	0.79	0.78	0.79	0.75	0.79	0.74
Submn. L	0.89	0.80	0.89	0.80	0.89	0.83	0.89	0.82	0.89	0.85	0.90	0.82
Submn R	0.85	0.80	0.81	0.80	0.85	0.84	0.87	0.83	0.85	0.87	0.88	0.84
Avg. ECE-label												
Deep Ensemble	0.07	0.07	0.05	0.05	0.07	0.16	0.07	0.17	0.07	0.11	0.07	0.10
MC-dropout	0.06	0.06	0.06	0.07	0.06	0.15	0.06	0.18	0.06	0.10	0.06	0.10

Ideally $F_{baseline}(D_{in}) = F_{OODmodel}(D_{in})$ (desideratum 1.1) and $F_{OODmodel}(D_{out}) < F_{OODmodel}(D_{in})$ (desideratum 1.2) hold true for the average surface DSC score. And $F_{baseline}(D_{in}) = F_{baseline}(D_{out}) = F_{OOD}(D_{in}) = F_{OOD}(D_{out})$ hold true for the ECE-label values (desideratum 2.1).

ID = in-distribution, OOD = out-of-distribution, Avg. = average, DSC = Sørensen Dice Similarity Coefficient, Buc = buccal, submn = submandibular, ECE = expected calibration error.

NOTE: e.g. for the no-metal-model, metal patients are out-of-distribution. The ECE-label value is computed, as described in chapter 5. Furthermore, note that e.g. the ID and OOD patients from the metal artefacts section (so both in the baseline model and the No-Metal model) are the same patients.

Analysis 3.1 Stable uncertainty computation for ID patients

Experiment 3.1 evaluated the stability of uncertainty computation for ID on the baseline and OOD models. Figure 6.5 shows that the no-bolus and no-bitelock models maintained stable uncertainty map metrics, comparable to the baseline on ID data. However, ID data on the no-metal model had substantially higher $N_{voxels, U>0.3}$ and higher mean uncertainty than on the baseline model. This is also observed in Figure 6.4, where the histogram for the no-metal model shows an increase in the number of voxels with uncertainty values between 0.05 and 0.4 on ID data.

Analysis 3.2 Higher uncertainty computation for OOD patients than ID patients on OOD models

Experiment 3.2 evaluated whether an OOD model showed higher uncertainty on OOD data than on ID data. Figure 6.4 reveals a clear distinction between OOD from ID patients for the no-metal and the no-bitelock model. With more voxels showing uncertainty above ± 0.05 for OOD patients than for ID patients. Furthermore, uncertainty values for both ID and OOD were higher in general for the no-metal model compared to the other OOD models. The no-bolus model showed a less visible difference in uncertainty histograms between OOD and ID data.

6.5 show a significantly higher $N_{voxels, U>0.3}$ and mean uncertainty values for OOD data than ID data on the no-metal model. This trend was observed with both Deep Ensemble

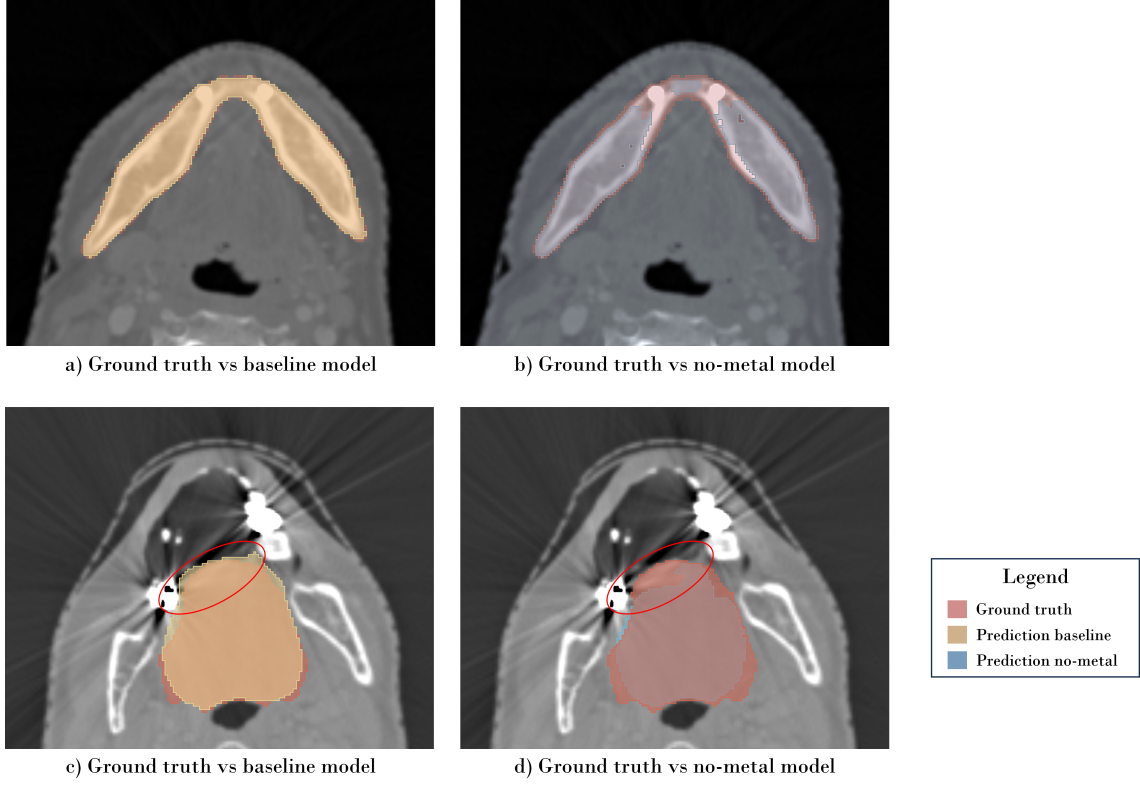


Figure 6.3: Segmentation prediction comparison between baseline and no-metal model for the mandible and oral cavity. TO DO: also show uncertainty for these slices

modelling and MC-dropout. The no-bolus model showed similar values for $N_{voxels,U>0.3}$ for ID and OOD data, while a statistically significant increase was observed in mean uncertainty for OOD data on the no-bitelock model with Deep Ensemble modelling but not with MC-dropout.

Analysis 3.3 Discrimination between OOD and ID data using a baseline versus an OOD model

Experiment 3.3 assessed whether OOD models outperform the baseline model in detecting OOD data. Figure 6.4 illustrates that the no-metal and no-bitelock models show less overlap between ID and OOD data in terms of $N_{voxels,U>0.3}$ compared to the baseline model. In other words, a greater distinction between OOD and ID data is observed with the OOD model than with the baseline model. In contrast, the no-bolus model shows no such distinction between ID and OOD data when evaluating $N_{voxels,U>0.3}$, similar to the baseline model.

When evaluating mean uncertainty, only the no-metal model demonstrates a larger difference between ID and OOD data compared to the baseline model. Although the no-bolus and no-bitelock models show differences in mean uncertainty between ID and OOD data, these differences are comparable to those seen with the baseline model.

AUC analysis in Table 6.2 shows that training without OOD data did not improve OOD detection for metal artefacts when using mean uncertainty. However, OOD detection did improve when using $N_{voxels,U>0.3}$ with MC-dropout. For bitelock patients, OOD detection improved with the OOD model, both when using mean uncertainty and $N_{voxels,U>0.3}$. In contrast, for bolus patients, training without OOD data did not enhance OOD detection with either uncertainty histogram metric.

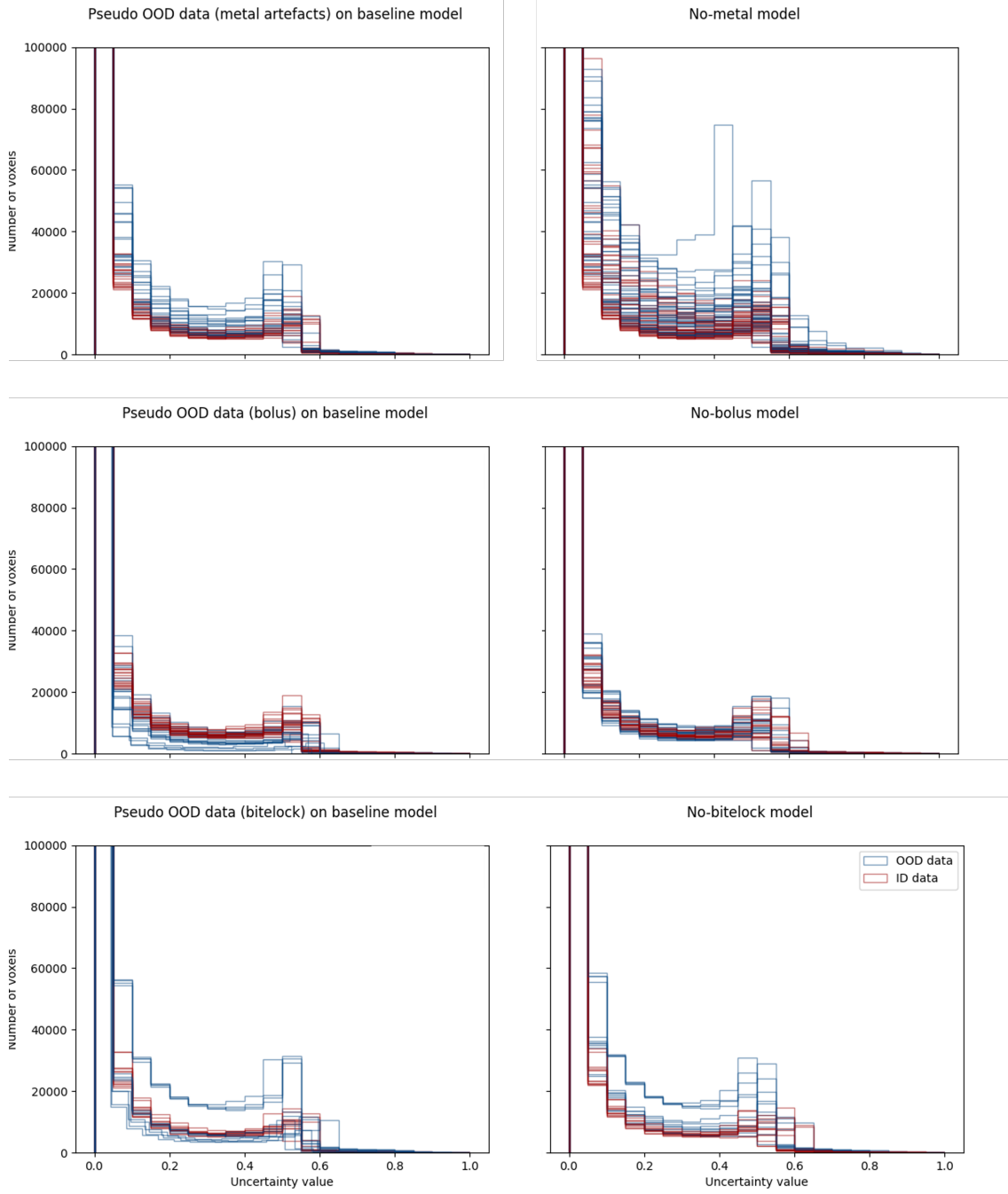
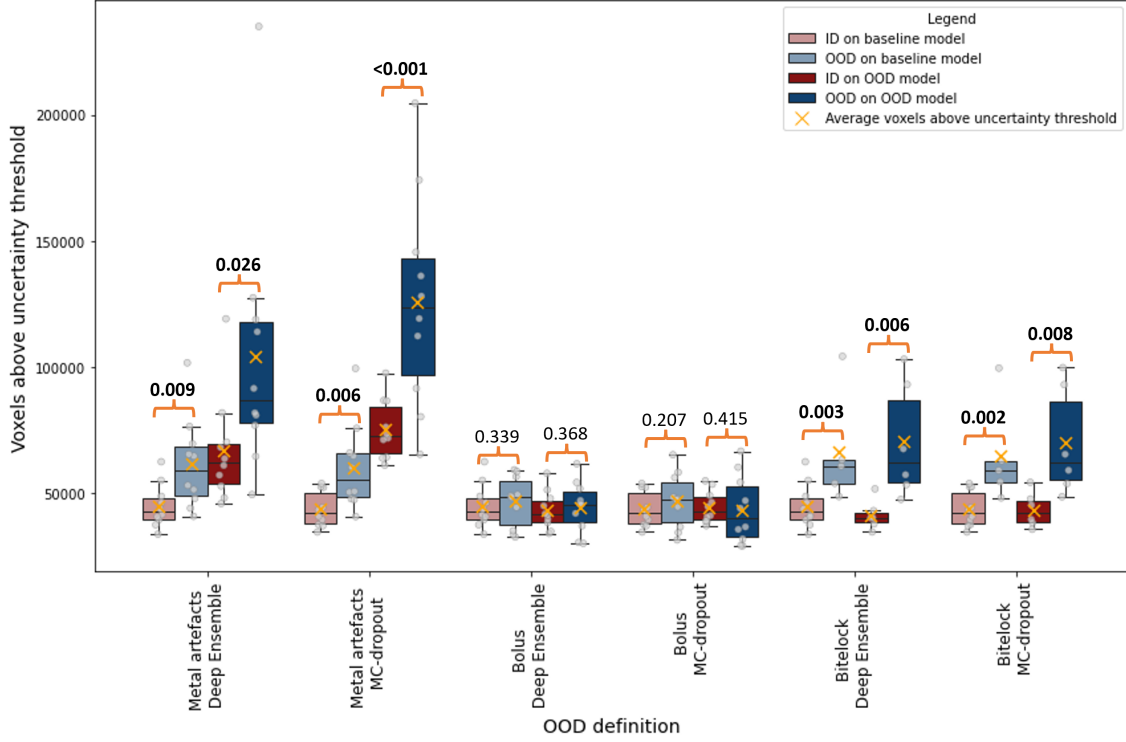
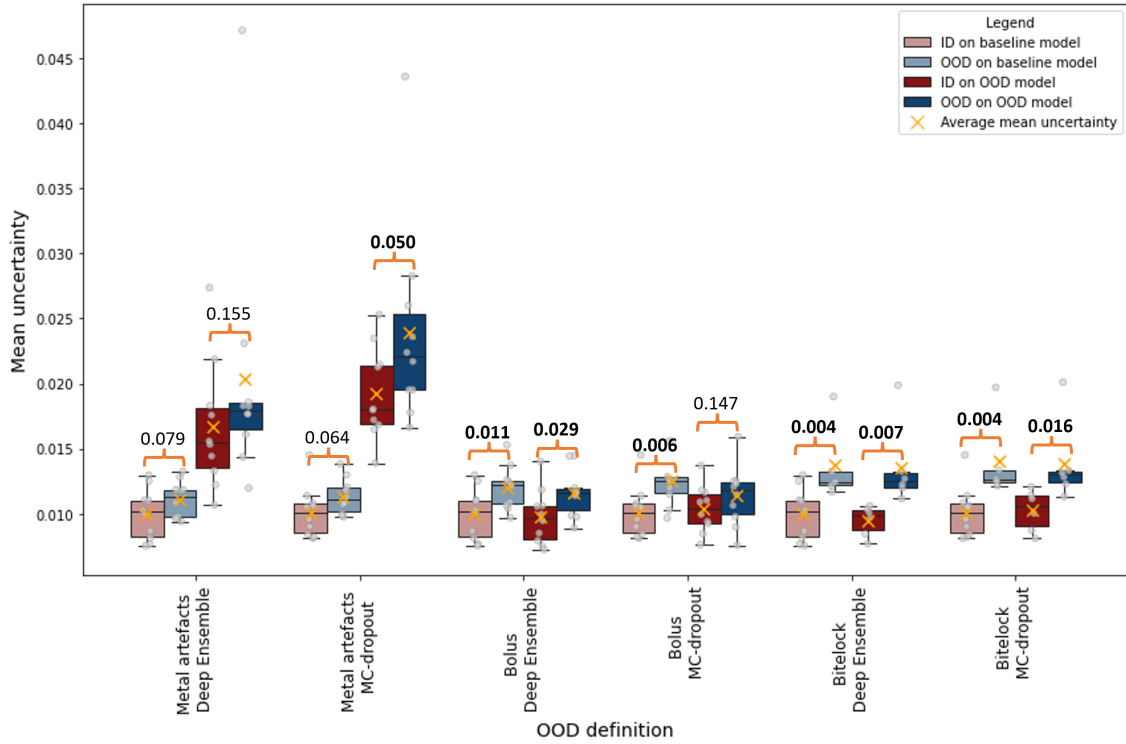


Figure 6.4: Uncertainty value histograms visualising ID vs OOD on no-metal-model (a), no-bolus model (b) and no-bitelock model (c). Uncertainty obtained with MC-dropout. Ideally $F_{OOD}(D_{out}) > F_{OOD}(D_{in})$ holds for higher uncertainty values (desideratum 3.2) Note: For improved visibility, the y-axis is truncated at $10e4$ to exclude background voxels with uncertainties below 0.05. Furthermore, note that ID data comprises the same cohort for all OOD models, but OOD data is selected per OOD model (e.g. metal artefact patients for the no-metal model).



a) Voxels above uncertainty threshold ($T = 0.3$)



b) Mean uncertainty

Figure 6.5: Comparison of uncertainty map metrics between ID and OOD data on the baseline model and OOD model. A distinction between uncertainty method (Deep Ensemble modelling and MC-dropout) is made.

Ideally $F_{baseline}(D_{in}) = F_{OOD}(D_{in})$ (desideratum 3.1), $F_{OOD}(D_{out}) > F_{OOD}(D_{in})$ (desideratum 3.2) and $F_{OOD}(D_{out}) - F_{OOD}(D_{in}) > F_{baseline}(D_{out}) - F_{baseline}(D_{in})$ (desideratum 3.3) hold true.

Note: ID data comprises the same cohort for all OOD definitions, but the OOD data is selected per OOD model (e.g. metal artefact patients for the no-metal model). Statistical significance is computed with an independent samples T-test where a p-value ≤ 0.05 is considered statistically significant and portrayed in bold.

Table 6.2: Model performance and uncertainty reliability analysis for ID and OOD on the baseline model against OOD models

	Metal artefacts				Bolus				Bitelock			
	Baseline model		No-metal model		Baseline model		No-bolus model		Baseline model		No-bitelock model	
	DE	MCD	DE	MCD	DE	MCD	DE	MCD	DE	MCD	DE	MCD
AUC												
Mean uncertainty	0.65	0.73	0.65	0.73	0.75	0.84	0.75	0.64	0.88	0.92	1.00	0.94
No. voxels $T > 0.3$	0.83	0.82	0.80	0.89	0.57	0.59	0.57	0.41	0.88	0.94	0.97	0.94

AUC analysis evaluates the ability of the uncertainty metric (mean uncertainty or number of voxels greater than 0.3) to perform OOD detection. AUC values for each OOD and uncertainty method are reported.

Ideally the OOD models have a higher AUC than the baseline model.

DE = Deep Ensemble modelling, MCD = MC-dropout

6.6 Discussion

There is currently no consensus on how to translate theoretically expected uncertainty behaviour into practical real-world application. Particularly for segmentation models in radiotherapy, there has been no investigation into uncertainty behaviour. This study evaluated the ability of various uncertainty methods- Deep Ensemble modelling, MC-dropout and Test-time augmentation- in distinguishing between ID and OOD data across three RT-specific scenarios: metal artefacts, bolus and bitelock. Our results showed that uncertainty quantification allows OOD detection. This was primarily the case for models trained using only ID data. Training on a combination of ID and OOD data reduced the value of uncertainty quantification for OOD detection. This indicates that uncertainty computation can effectively reflect knowledge gaps in a model caused by a lack of representation of the test case in the training distribution.

6.6.1 Model performance

In this study, we observed that excluding certain patient subsets, such as those with metal artefacts, resulted in a degradation of performance for specific OARs in the ID dataset. Notably, structures like the right buccal mucosa, oral cavity, and right submandibular gland on the ID patients experienced a reduction in segmentation accuracy when trained on a restricted dataset without metal artefacts.

For the right buccal mucosa, the decrease in performance was primarily driven by one outlier patient whose right buccal mucosa extended around the teeth. This presented a challenge for the no-metal model, which had less exposure to teeth, as most patients with teeth were excluded due to having metal fillings. The baseline model, which included patients with metal artefacts in various locations, likely performed better due to the broader range of patients included. Additionally, the observed reduction in OOD model performance on ID data may also stem from differences in image preprocessing: the mean CT intensity was 153 Hounsfield unit (HU) for the baseline model and -2 HU for the no-metal model. This variation in CT normalization could have affected the visibility of structures influenced by metal artefacts, such as the oral cavity and submandibular gland. These decreases in performance, while within the observed standard deviation of Chapter 4, underscore that excluding a subset of patients can also negatively impact model generalisation on ID patients as this reduces the comprehensiveness of the training distribution, thus indicating the importance of including a broad training set.

Secondly, model performance was also compared between ID and OOD data on the OOD model and baseline model. A drop in performance on OOD patients in the OOD model would indicate that the knowledge of the OOD model was successfully compromised by

selectively limiting the training data distribution, especially if performance on the OOD model did not decrease to the same extent as the baseline model. This was observed for the mandible and oral cavity on the no-metal model, where performance reduced to below the standard deviation observed in chapter 4. The decreased performance in these OARs aligns with expectations, as these structures are closest to the metal artefacts, as shown in Figure 6.3, confirming that the model had limited exposure to such artefacts.

For bitelock OOD patients, a decrease in performance was observed across all OARs, with similar effects seen in both the baseline and no-bitelock models. Although this reduction was near the boundary of the standard deviation from Chapter 4 for the left and right buccal mucosa and the left parotid gland, it suggests that bitelock affects the positioning and shape of the mouth. The consistent performance drop in both the OOD and baseline models highlights that bitelock patients represent a relatively small subset of the population, affecting model generalizability.

The observations affirm that restricting the training data distribution has influenced the model’s performance on an OOD dataset and that it is thus interesting to see if uncertainty computation reflects this behaviour.

6.6.2 Uncertainty reliability

For experiment 2.1 the reliability of the uncertainty values was evaluated. For ID patients all models showed good uncertainty value calibration with ECE-label values between 0.06 and 0.07 for all models and both Deep Ensemble modelling and MC-dropout. For metal artefact OOD patients the model also showed good calibration both for the baseline and no-metal model. However for bolus and bitelock patients, ECE-label value increased substantially for both the baseline and OOD models. Analysing the reliability diagrams showed no pattern as both over- and underconfidence were observed.

6.6.3 Uncertainty for OOD detection

The results from experiments 3.1 to 3.3 demonstrate that Deep Ensemble modelling shows superior performance in detecting far-OOD bitelock patients on the no-bitelock model, achieving an AUC of 1.00 using the mean uncertainty. The baseline model also achieved strong OOD detection performance, with AUC scores ranging from 0.88 to 0.94 on both the mean uncertainty and $N_{voxels,U>0.3}$ for Deep Ensemble modelling and MC-dropout. The high AUC scores for the baseline model can be attributed to the small representation of bitelock patients in the complete training population for the baseline model. However excluding the subset does improve OOD detection. This indicates that uncertainty is an effective metric to evaluate model knowledge. The observation that Deep Ensemble modelling performs best for far-OOD detection is in line with previous findings from OOD detection tasks by Linmans et al. [105]. AUC scores for the bolus definition of OOD did not improve nor did the uncertainty histograms metrics show the ability to discriminate between ID and OOD. This suggests that a bolus did not influence the model’s knowledge and was thus an ineffective definition of OOD as was also indicated by the lack of reduction in model performance compared to ID data for this definition of OOD.

The results also demonstrate that uncertainty computation is capable of detecting shifted-OOD metal artefact patients on the no-metal model. The no-metal model outperforms the baseline model as demonstrated by a greater difference between ID and OOD data for $N_{voxels,U>0.3}$, resulting in a higher AUC score of 0.89 for OOD detection with MC-dropout. However, the baseline model also performs well, with AUC scores of 0.83 and 0.82 for Deep Ensemble modelling and MC-dropout, respectively, indicating that it also has uncertainty on metal artefact data, likely due to the wide variety of possible metal artefacts and

the limited number of training data in the baseline model. Deep Ensemble modelling on the no-metal model performed inferior to MC-dropout with an AUC of 0.80. Mean uncertainty yielded significantly lower AUC values of 0.65 and 0.73 for Deep Ensemble modelling and MC-dropout in both models, with identical results. Despite this, uncertainty histograms in Figure 6.4 suggest that Deep Ensemble modelling has the potential for effective discrimination. Future research could explore alternative uncertainty histogram metrics that investigate relative uncertainty values per individual data subject (e.g. thresholding based on the 5th percentile), as proposed by Linmans et al. [105].

The inconsistent superiority of Deep Ensemble modelling and MC-dropout across different OOD scenarios suggest that the effectiveness of uncertainty methods is application-specific, necessitating performance evaluations per use-case.

6.6.4 Limitations

A limitation of this study is that analysis was performed on the entire CT-volume. Given that artefacts cause localised distortions, aggregating uncertainty into a single metric, as was performed, might overlook these nuances. While patient-level OOD definitions justified this approach, future research could explore slice-level or another form of local OOD definitions to enhance detection performance and practical usability.

Additionally, the OOD definitions used in this study are subjective and potentially biased. For example, a bolus is typically applied for patients having tumours near the skin surface, which creates a distinct patient group of with possible different tumour location and TNM staging compared to the standard patient group. This introduces confounding factors beyond the bolus itself. Secondly, while the current OOD definitions have limited practical value, as they are easily distinguished by humans, this study establishes a foundation for evaluating the influence of uncertainty on OOD detection in RT medical imaging data. Future research can concern more practical values of unknown OOD for clinical translation. An interesting example would for example consider data from different institutions as OOD. This was proposed in this study, however, due to limited time, the OOD model and uncertainty maps were only developed but not evaluated.

Models were cross-validation trained with a limited validation set of 12 patients. To the best of our knowledge, the validation set is only used to report model performance during training and select the best-performing model per fold. Furthermore, the model was also tested on a limited patient set (20 for metal artefacts and bolus patients and 12 for bitelock patients). Although promising results for OOD detection were observed with high AUC values, it is important to keep the limited dataset in mind. A future study with a more substantial test set is necessary.

Unfortunately, due to limited time, Test-time augmentation was not implemented. Theoretically, Test-time augmentation visualises aleatoric uncertainty [37], the randomness in the data itself. It should be uninfluenced by OOD data. However, disentangling uncertainty in practice is not straightforward. [72] Including Test-time augmentation and a comparison with the predictive uncertainty methods MC-dropout and Deep Ensemble modelling in future research would provide valuable insights into the value and possibility of uncertainty disentangling for OOD detection.

Overall, this study demonstrated the ability of uncertainty to perform OOD detection in RT. However, the true value of uncertainty for OOD lies in the clinical application. We did not evaluate whether uncertainty and OOD detection would influence clinician trust and behaviour, or how the automated RT workflow could benefit from uncertainty for OOD detection. We leave the clinical translation of uncertainty for future research.

6.7 Conclusion

This study evaluated the ability of two popular uncertainty methods- Deep Ensemble modelling and MC-dropout - in performing OOD detection in segmentation tasks. The experiments targeted expected uncertainty behaviour in realistic clinical challenges in RT, including metal artefacts (shifted-OOD) and bitelock and bolus patients (far-OOD). We observed that metal artefacts and bitelock presence effectively compromised model knowledge, with MC-dropout-based uncertainty showing a strong capability of discriminating between ID and OOD (AUC 0.89). Deep Ensemble modelling-based uncertainty showed superior performance in performing far-OOD on bitelock patients (AUC 1.00). These results show the potential of uncertainty to perform OOD detection, but also the importance of problem-dependent evaluation of uncertainty computation methods. Uncertainty for OOD detection could assist QA in an automated/adaptive RT workflow, helping to identify cases that fall outside of the model's knowledge that requires manual evaluation. Future research could explore the ability of uncertainty computation in more localised definitions of OOD and assess the clinical effect of uncertainty information.

Chapter 7

General conclusion and discussion

This research evaluated the reliability of uncertainty computation and its ability to assess the real-world generalisability of a segmentation model in RT. Using nnU-Net, we developed a high-performing DL segmentation network for 19HNC OARs based on CT-images. We implemented and evaluated the reliability of three popular uncertainty computation methods: Deep Ensemble modelling, MC-dropout and Test-time augmentation. Wahid et al. [33] have shown that consensus on uncertainty method and metric is lacking when applying uncertainty computation in RT. We have demonstrated that among the most popular uncertainty methods, the specific method itself has a negligible impact compared to the choice of uncertainty metric. Entropy showed superior performance with well-calibrated uncertainty values for all uncertainty methods and stable model performance, on multi-class and classwise levels. In contrast, variance and mutual information showed poor calibration. This underscores the importance of carefully selecting uncertainty metrics and evaluating calibration.

In theory, (disentangled) uncertainty quantification should reflect a model’s confidence concerning the completeness of the training data distribution. [32] However, there is no unified consensus on its practical use in RT, with some attributing it to aleatoric uncertainty [40, 41], and others to epistemic uncertainty [39, 7, 42]. We shifted beyond this theoretical discourse and focused on an empirical evaluation of the ability of uncertainty computation to identify the limitations of a segmentation model’s generalisability. We implemented entropy on Deep Ensemble modelling and MC-dropout and defined OOD detection problems on real-world examples from the RT domain using patients with metal artefacts, bitelock and a bolus as variants in the data distribution. The model’s knowledge and segmentation performance for specific OARs near metal artefacts and bitelocks were degraded when removing these data subsets from the training distribution.

Uncertainty computation effectively discriminated between ID and OOD data, with AUCs of 0.89 for metal artefacts and 1.00 for bitelock patients. This validates the theoretical premise that uncertainty computation can effectively capture the limitations of a model’s knowledge when applied to data outside of the training environment in a real-world radiotherapy scenario. This is valuable in an automated RT workflow, such as adaptive RT, where efficient relevant adjustments are of interest. Uncertainty in OOD detection could guide the identification of cases outside the model’s knowledge domain, allowing for targeted manual evaluation or model improvement—ultimately enhancing patient care and reducing radiation toxicity.

Chapter 8

Future research

This thesis explored the reliability of uncertainty in segmentation for RT, an area currently without consensus on a single method and metric for uncertainty computation, as highlighted by Wahid et al. [33]. Notably, this challenge of establishing a standardised method and metric presents to be not limited to RT, but other domains in medical imaging and machine learning as well. Our research, like foregoing research, focused on a highly targeted application case: HNC OAR segmentation. Consequently, the findings do not contribute to resolving the general lack of agreement in the field. Nonetheless, we presented a method to thoroughly evaluate uncertainty computation that can be valuable in comparing uncertainty computation quality between studies. Moving forward, it remains necessary to perform a comprehensive and standardised evaluation of uncertainty computation quality across different segmentation and classification application in the medical imaging domain. The diverse performance results for uncertainty metrics and methods observed in this study further underscore the need for a standardised, case-by-case evaluation of uncertainty computation quality including a reliability analysis before implementation.

Furthermore, our research focused on a convolution-based nnU-Net segmentation network. Evaluating uncertainty computation across different models could be very interesting, for instance by comparing a transformer-based network, such as Swin-UNETR [110], with a traditional U-Net segmentation model. Transformer networks have demonstrated promising results in medical imaging segmentation tasks [110] outperforming the nnU-Net, and it could be anticipated that model and predictive uncertainty computation might manifest differently between convolutional and transformer-based models due to their distinct approaches to extracting knowledge from medical images. Investigating these differences could offer a deeper understanding of how uncertainty varies with different model architectures and what it reveals. Roshanzamir et al. [106] have already observed differences in uncertainty computation between transformer-based and U-Net models, however, their study focused on the relationship between uncertainty and inter-rater variability- an uncertainty aspect generally considered as arising from the data and not from the model. It is therefore interesting for future research to investigate the influence of different segmentation models on the usability of uncertainty for OOD detection or another form of model uncertainty.

Similarly to new segmentation models, the field of uncertainty computation is continuously evolving. Emerging methods such as evidential deep learning [111] and deep deterministic uncertainty [104] offer promising new approaches for determining uncertainty. As we focused on the three most popular uncertainty methods, these new methods were not evaluated in this study. However, for future research, it would be interesting to include a broader range of uncertainty methods.

To further explore the idea of uncertainty and disentanglement, it is essential to apply (the same) uncertainty computation on several tasks, preferably in a single study. While our research focused exclusively on OOD, investigating inter- and intra-observer variability with the same methods could offer deeper insights into what uncertainty visualises and if it disentangles. Additionally, training a model using multiple instances of the same patients with different variants of manual segmentation could help examine how these data variations influence the model’s performance and uncertainty.

Furthermore, we defined our criteria for OOD data to evaluate the real-world data distribution for an RT segmentation task. However, a more quantitative and potentially more accurate approach would involve establishing methods to measure model generalisability. This could for example be achieved through assessing or predicting dataset generalisability. [112]. Another option would be to perform a data analysis (e.g. shape, volume, or intensity distribution of the data), allowing for a more objective definition of OOD.

Finally, the clinical implementation of uncertainty in RT must remain a priority when conducting uncertainty research. Since uncertainty is intended to aid in semi-manual QA within an automated RT workflow, it is essential to consider its practical value for clinicians. Keeping them involved ensures that the methods developed address actual clinical needs. The ultimate goal is for uncertainty to guide useful adjustments. In RT, an adjustment is valuable if it impacts the radiation dose delivered to the patient. Thus, the future of uncertainty research in segmentation for RT lies at the convergence of reliable uncertainty computation, segmentation, and treatment planning.

Bibliography

- [1] Luther W Brady et al. *Functional preservation and quality of life in head and neck radiotherapy*. Springer, 2009.
- [2] Athanassios Argiris et al. “Head and neck cancer”. In: *The Lancet* 371.9625 (2008), pp. 1695–1709.
- [3] Garrett Anderson et al. “An updated review on head and neck cancer treatment with radiation therapy”. In: *Cancers* 13.19 (2021), p. 4912.
- [4] Peiru Liu et al. “Deep learning algorithm performance in contouring head and neck organs at risk: a systematic review and single-arm meta-analysis”. In: *BioMedical Engineering OnLine* 22.1 (2023), p. 104.
- [5] C Robert et al. “Clinical implementation of deep-learning based auto-contouring tools—Experience of three French radiotherapy centers”. In: *Cancer/Radiothérapie* 25.6-7 (2021), pp. 607–616.
- [6] Lisanne V Van Dijk et al. “Improving automatic delineation for head and neck organs at risk by Deep Learning Contouring”. In: *Radiotherapy and Oncology* 142 (2020), pp. 115–123.
- [7] Ana Barragán-Montero et al. “Towards a safe and efficient clinical implementation of machine learning in radiation oncology by exploring model interpretability, explainability and data-model dependency”. In: *Physics in Medicine & Biology* 67.11 (2022), 11TR01.
- [8] Patrick J González et al. “Explaining the dosimetric impact of contouring errors in head and neck radiotherapy”. In: *Biomedical Physics & Engineering Express* 8.5 (2022), p. 055001.
- [9] Terrance DeVries and Graham W Taylor. “Leveraging uncertainty estimates for predicting segmentation quality”. In: *arXiv preprint arXiv:1807.00502* (2018).
- [10] Carlos E Cardenas et al. “Advances in auto-segmentation”. In: *Seminars in radiation oncology*. Vol. 29. 3. Elsevier. 2019, pp. 185–197.
- [11] Charlotte L Brouwer et al. “Assessment of manual adjustment performed in clinical practice following deep learning contouring for head and neck organs at risk in radiotherapy”. In: *Physics and imaging in radiation oncology* 16 (2020), pp. 54–60.
- [12] Ying Zhang et al. “Texture-based, automatic contour validation for online adaptive replanning: a feasibility study on abdominal organs”. In: *Medical physics* 46.9 (2019), pp. 4010–4020.
- [13] Shunyao Luan et al. “Machine Learning-Based Quality Assurance for Automatic Segmentation of Head-and-Neck Organs-at-Risk in Radiotherapy”. In: *Technology in Cancer Research & Treatment* 22 (2023), p. 15330338231157936.

- [14] Hsin-Chen Chen et al. “Automated contouring error detection based on supervised geometric attribute distribution models for radiation therapy: a general strategy”. In: *Medical physics* 42.2 (2015), pp. 1048–1059.
- [15] MB Altman et al. “A framework for automated contour quality assurance in radiation therapy including adaptive techniques”. In: *Physics in Medicine & Biology* 60.13 (2015), p. 5199.
- [16] Cheukkai B Hui et al. “Quality assurance tool for organ at risk delineation in radiation therapy using a parametric statistical approach”. In: *Medical physics* 45.5 (2018), pp. 2089–2096.
- [17] Jingwei Duan et al. “Contouring quality assurance methodology based on multiple geometric features against deep learning auto-segmentation”. In: *Medical Physics* (2023).
- [18] Zachary T Wooten et al. “Predictive modeling using shape statistics for interpretable and robust quality assurance of automated contours in radiation treatment planning”. In: *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2023: Kohala Coast, Hawaii, USA, 3–7 January 2023*. World Scientific. 2022, pp. 395–406.
- [19] Chris McIntosh, Igor Svistoun, and Thomas G Purdie. “Groupwise conditional random forests for automatic shape classification and contour quality assessment in radiotherapy planning”. In: *IEEE transactions on medical imaging* 32.6 (2013), pp. 1043–1057.
- [20] Pavlos Papaconstadopoulos et al. “An anomaly detector as a clinical decision support system for parotid gland delineations”. In: *Physics in Medicine & Biology* 66.10 (2021), p. 105017.
- [21] Hamidreza Nourzadeh et al. “Knowledge-based quality control of organ delineations in radiation therapy”. In: *Medical physics* 49.3 (2022), pp. 1368–1381.
- [22] Dong Joo Rhee et al. “Automatic detection of contouring errors using convolutional neural networks”. In: *Medical physics* 46.11 (2019), pp. 5086–5097.
- [23] Kuo Men et al. “Automated quality assurance of OAR contouring for lung cancer based on segmentation with deep active learning”. In: *Frontiers in Oncology* 10 (2020), p. 986.
- [24] Hang Min et al. “Clinical target volume delineation quality assurance for MRI-guided prostate radiotherapy using deep learning with uncertainty estimation”. In: *Radiotherapy and Oncology* 186 (2023), p. 109794.
- [25] Michaël Claessens et al. “Machine learning-based detection of aberrant deep learning segmentations of target and organs at risk for prostate radiotherapy using a secondary segmentation algorithm”. In: *Physics in Medicine & Biology* 67.11 (2022), p. 115014.
- [26] Vanya V Valindria et al. “Reverse classification accuracy: predicting segmentation performance in the absence of ground truth”. In: *IEEE transactions on medical imaging* 36.8 (2017), pp. 1597–1606.
- [27] Robert Robinson et al. “Subject-level prediction of segmentation failure using real-time convolutional neural nets”. In: (2018).
- [28] Xinyuan Chen et al. “CNN-based quality assurance for automatic segmentation of breast cancer in radiotherapy”. In: *Frontiers in Oncology* 10 (2020), p. 524.
- [29] Chong Su et al. “DEEP LEARNING APPROACHES FOR ANOMALIES DETECTION OF BLADDER CT CONTOURS IN PROSTATE CANCER PATIENTS”. In: *Journal of Mechanics in Medicine and Biology* 22.08 (2022), p. 2240033.

- [30] David JC MacKay. “A practical Bayesian framework for backpropagation networks”. In: *Neural computation* 4.3 (1992), pp. 448–472.
- [31] Radford M Neal. *Bayesian learning for neural networks*. Vol. 118. Springer Science & Business Media, 2012.
- [32] Yarin Gal et al. “Uncertainty in deep learning”. In: (2016).
- [33] Kareem A Wahid et al. “Harnessing uncertainty in radiotherapy auto-segmentation quality assurance”. In: *Physics and Imaging in Radiation Oncology* 29 (2024).
- [34] Yarin Gal and Zoubin Ghahramani. “Dropout as a bayesian approximation: Representing model uncertainty in deep learning”. In: *international conference on machine learning*. PMLR. 2016, pp. 1050–1059.
- [35] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. “Simple and scalable predictive uncertainty estimation using deep ensembles”. In: *Advances in neural information processing systems* 30 (2017).
- [36] Murat Seckin Ayhan and Philipp Berens. “Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks”. In: *Medical Imaging with Deep Learning*. 2022.
- [37] Guotai Wang et al. “Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks”. In: *Neurocomputing* 338 (2019), pp. 34–45.
- [38] Alex Kendall and Yarin Gal. “What uncertainties do we need in bayesian deep learning for computer vision?” In: *Advances in neural information processing systems* 30 (2017).
- [39] Cornelis AT van den Berg and Ettore F Meliador. “Uncertainty assessment for deep learning radiotherapy applications”. In: *Seminars in Radiation Oncology*. Vol. 32. 4. Elsevier. 2022, pp. 304–318.
- [40] Lotta Meijerink, Giovanni Cinà, and Michele Tonutti. “Uncertainty estimation for classification and risk prediction on medical tabular data”. In: *arXiv preprint arXiv:2004.05824* (2020).
- [41] Niclas Ståhl et al. “Evaluation of uncertainty quantification in deep learning”. In: *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer. 2020, pp. 556–568.
- [42] Maria Leousi. “Estimating uncertainty of deep learning-based segmentation for prostate cancer radiotherapy”. In: (2017).
- [43] Fabian Isensee et al. “nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation”. In: *Nature methods* 18.2 (2021), pp. 203–211.
- [44] Murat Beyzadeoglu, Gokhan Ozyigit, and Cuneyt Ebruli. *Basic radiation oncology*. Vol. 71. Springer, 2020.
- [45] A.J.J Bos, F.S. Draaisma, and W.J.C Okx. *Inleiding tot de stralingshygiëne*. Sdu Uitgevers bv, Den Haag, 2007.
- [46] Charlotte L Brouwer et al. “CT-based delineation of organs at risk in the head and neck region: DAHANCA, EORTC, GORTEC, HKNPCSG, NCIC CTG, NCRI, NRG Oncology and TROG consensus guidelines”. In: *Radiotherapy and Oncology* 117.1 (2015), pp. 83–90.
- [47] Edvard Abel et al. “Long-term aspects of quality of life in head and neck cancer patients treated with intensity modulated radiation therapy: a 5-year longitudinal follow-up and comparison with a normal population cohort”. In: *Advances in radiation oncology* 5.1 (2020), pp. 101–110.

- [48] Xiaojing Yang et al. “Radiation-induced skin injury: pathogenesis, treatment, and management”. In: *Aging (Albany NY)* 12.22 (2020), p. 23379.
- [49] Itzhak Brook. “Late side effects of radiation treatment for head and neck cancer”. In: *Radiation oncology journal* 38.2 (2020), p. 84.
- [50] Daniela Alterio et al. “Modern radiotherapy for head and neck cancer”. In: *Seminars in oncology*. Vol. 46. 3. Elsevier. 2019, pp. 233–245.
- [51] Kush Panara, Edris R. Ahanga, and Devang Ppadalia. *Physiology, Swallowing StatPearls*. 2021. URL: <https://www.ncbi.nlm.nih.gov/books/NBK541071/> (visited on 07/19/2024).
- [52] Julie Van der Veen and Sandra Nuyts. “Can intensity-modulated-radiotherapy reduce toxicity in head and neck squamous cell carcinoma?” In: *Cancers* 9.10 (2017), p. 135.
- [53] Katherine A Hutcheson et al. “Late dysphagia after radiotherapy-based treatment of head and neck cancer”. In: *Cancer* 118.23 (2012), pp. 5793–5799.
- [54] Herve Y Sroussi et al. “Common oral complications of head and neck cancer radiation therapy: mucositis, infections, saliva change, fibrosis, sensory dysfunctions, dental caries, periodontal disease, and osteoradionecrosis”. In: *Cancer medicine* 6.12 (2017), pp. 2918–2931.
- [55] Ling Zhou et al. “Research progress of radiation-induced hypothyroidism in head and neck cancer”. In: *Journal of Cancer* 12.2 (2021), p. 451.
- [56] Pranav Rajpurkar et al. “AI in health and medicine”. In: *Nature medicine* 28.1 (2022), pp. 31–38.
- [57] Lars Johannes Isaksson et al. “Automatic Segmentation with Deep Learning in Radiotherapy”. In: *Cancers* 15.17 (2023), p. 4389.
- [58] Tomaž Vrtovec et al. “Auto-segmentation of organs at risk for head and neck radiotherapy planning: from atlas-based to deep learning methods”. In: *Medical physics* 47.9 (2020), e929–e950.
- [59] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).
- [60] William Fedus, Barret Zoph, and Noam Shazeer. “Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity”. In: *Journal of Machine Learning Research* 23.120 (2022), pp. 1–39.
- [61] Geert Litjens et al. “A survey on deep learning in medical image analysis”. In: *Medical image analysis* 42 (2017), pp. 60–88.
- [62] S Kevin Zhou, Hayit Greenspan, and Dinggang Shen. *Deep learning for medical image analysis*. Academic Press, 2023.
- [63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III* 18. Springer. 2015, pp. 234–241.
- [64] Guillaume Landry, Christopher Kurz, and Alberto Traverso. “The role of artificial intelligence in radiotherapy clinical practice”. In: *BJR— Open* 5.1 (2023), p. 20230030.
- [65] Dong Joo Rhee et al. “Automatic contouring QA method using a deep learning-based autocontouring system”. In: *Journal of applied clinical medical physics* 23.8 (2022), e13647.

- [66] Laurent Valentin Jospin et al. “Hands-on Bayesian neural networks—A tutorial for deep learning users”. In: *IEEE Computational Intelligence Magazine* 17.2 (2022), pp. 29–48.
- [67] Jishnu Mukhoti and Yarin Gal. “Evaluating bayesian deep learning methods for semantic segmentation”. In: *arXiv preprint arXiv:1811.12709* (2018).
- [68] Thomas G Dietterich. “Ensemble methods in machine learning”. In: *International workshop on multiple classifier systems*. Springer. 2000, pp. 1–15.
- [69] Moloud Abdar et al. “A review of uncertainty quantification in deep learning: Techniques, applications and challenges”. In: *Information fusion* 76 (2021), pp. 243–297.
- [70] Matias Valdenegro-Toro and Daniel Saromo Mori. “A deeper look into aleatoric and epistemic uncertainty disentanglement”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE. 2022, pp. 1508–1516.
- [71] Tanya Nair et al. “Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation”. In: *Medical image analysis* 59 (2020), p. 101557.
- [72] Lisa Wimmer et al. “Quantifying aleatoric and epistemic uncertainty in machine learning: Are conditional entropy and mutual information appropriate measures?” In: *Uncertainty in Artificial Intelligence*. PMLR. 2023, pp. 2282–2292.
- [73] Armen Der Kiureghian and Ove Ditlevsen. “Aleatory or epistemic? Does it matter?” In: *Structural safety* 31.2 (2009), pp. 105–112.
- [74] Bertrand Charpentier et al. “Disentangling epistemic and aleatoric uncertainty in reinforcement learning”. In: *arXiv preprint arXiv:2206.01558* (2022).
- [75] Connor Shorten and Taghi M Khoshgoftaar. “A survey on image data augmentation for deep learning”. In: *Journal of big data* 6.1 (2019), pp. 1–48.
- [76] Shi Hu et al. “Supervised uncertainty quantification for segmentation with multiple annotations”. In: *Medical Image Computing and Computer Assisted Intervention—MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer. 2019, pp. 137–145.
- [77] Alain Jungo et al. “Uncertainty-driven sanity check: Application to postoperative brain tumor cavity segmentation”. In: *arXiv preprint arXiv:1806.03106* (2018).
- [78] Abhijit Guha Roy et al. “Bayesian QuickNAT: Model uncertainty in deep whole-brain segmentation for structure-wise quality control”. In: *NeuroImage* 195 (2019), pp. 11–22.
- [79] Alain Jungo, Fabian Balsiger, and Mauricio Reyes. “Analyzing the quality and challenges of uncertainty estimations for brain tumor segmentation”. In: *Frontiers in neuroscience* 14 (2020), p. 501743.
- [80] Ward van Rooij et al. “Deep learning-based delineation of head and neck organs at risk: geometric and dosimetric evaluation”. In: *International Journal of Radiation Oncology* Biology* Physics* 104.3 (2019), pp. 677–684.
- [81] Jaakko Sahlsten et al. “Application of simultaneous uncertainty quantification and segmentation for oropharyngeal cancer use-case with Bayesian deep learning”. In: *Communications Medicine* 4.1 (2024), p. 110.
- [82] Roque Rodríguez Outeiral et al. “A network score-based metric to optimize the quality assurance of automatic radiotherapy target segmentations”. In: *Physics and Imaging in Radiation Oncology* 28 (2023), p. 100500.

- [83] Charlotte L Brouwer et al. “Machine learning applications in radiation oncology: Current use and needs to support clinical implementation”. In: *Physics and imaging in radiation oncology* 16 (2020), pp. 144–148.
- [84] Tobias Heimann et al. “Comparison and evaluation of methods for liver segmentation from CT datasets”. In: *IEEE transactions on medical imaging* 28.8 (2009), pp. 1251–1265.
- [85] Jia, Jingnan. *Segmentation Metrics*. Version 1.1.6. Dec. 25, 2023.
- [86] Stanislav Nikolov et al. “Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy”. In: *arXiv preprint arXiv:1809.04430* (2018).
- [87] Nikolov, Stanislav. *Surface distance metrics*. Version 1.1.0. Dec. 25, 2023.
- [88] Femke Vaassen et al. “Evaluation of measures for assessing time-saving of automatic organ-at-risk segmentation in radiotherapy”. In: *Physics and Imaging in Radiation Oncology* 13 (2020), pp. 1–6.
- [89] Kiser, Kendall. *Autosegmentation Spatial Similarity Metrics*. Version 1.1.0. Dec. 25, 2023.
- [90] Michael V Sherer et al. “Metrics to evaluate the performance of auto-segmentation for radiation treatment planning: A critical review”. In: *Radiotherapy and Oncology* 160 (2021), pp. 185–191.
- [91] Joëlle van Aalst et al. “Clarifying the Dosimetric Impact of Autosegmentation Inaccuracies”. In: *International Conference on Computers in Radiotherapy (ICCR) 2024*. 2024.
- [92] Luke Whitbread and Mark Jenkinson. “Uncertainty categories in medical image segmentation: a study of source-related diversity”. In: *International Workshop on Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*. Springer. 2022, pp. 26–35.
- [93] Robin Camarasa et al. “Quantitative comparison of monte-carlo dropout uncertainty measures for multi-class segmentation”. In: *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Graphs in Biomedical Image Analysis: Second International Workshop, UNSURE 2020, and Third International Workshop, GRAIL 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 8, 2020, Proceedings 2*. Springer. 2020, pp. 32–41.
- [94] Yarín Gal, Jiri Hron, and Alex Kendall. “Concrete dropout”. In: *Advances in neural information processing systems* 30 (2017).
- [95] Lewis Smith and Yarín Gal. “Understanding measures of uncertainty for adversarial example detection”. In: *arXiv preprint arXiv:1803.08533* (2018).
- [96] Elisa Chotzoglou and Bernhard Kainz. “Exploring the relationship between segmentation uncertainty, segmentation performance and inter-observer variability with probabilistic networks”. In: *Large-Scale Annotation of Biomedical Data and Expert Label Synthesis and Hardware Aware Learning for Medical Imaging and Computer Assisted Intervention: International Workshops, LABELS 2019, HAL-MICCAI 2019, and CuRIOUS 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13 and 17, 2019, Proceedings 4*. Springer. 2019, pp. 51–60.
- [97] Katharina Hoebel et al. “An exploration of uncertainty information for segmentation quality assessment”. In: *Medical Imaging 2020: Image Processing*. Vol. 11313. SPIE. 2020, pp. 381–390.

- [98] Ishaan Bhat et al. “Influence of uncertainty estimation techniques on false-positive reduction in liver lesion detection”. In: *arXiv preprint arXiv:2206.10911* (2022).
- [99] Jörg Sander et al. “Towards increased trustworthiness of deep learning segmentation methods on cardiac MRI”. In: *Medical imaging 2019: image Processing*. Vol. 10949. SPIE. 2019, pp. 324–330.
- [100] Morris H DeGroot and Stephen E Fienberg. “The comparison and evaluation of forecasters”. In: *Journal of the Royal Statistical Society: Series D (The Statistician)* 32.1-2 (1983), pp. 12–22.
- [101] Chuan Guo et al. “On calibration of modern neural networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 1321–1330.
- [102] Mahdi Pakdaman Naeini, Gregory F Cooper, and Milos Hauskrecht. “Binary classifier calibration: Non-parametric approach”. In: *arXiv preprint arXiv:1401.3390* (2014).
- [103] Zhilu Zhang, Adrian V Dalca, and Mert R Sabuncu. “Confidence calibration for convolutional neural networks using structured dropout”. In: *arXiv preprint arXiv:1906.09551* (2019).
- [104] Jishnu Mukhoti et al. “Deep deterministic uncertainty: A new simple baseline”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 24384–24394.
- [105] Jasper Linmans et al. “Predictive uncertainty estimation for out-of-distribution detection in digital pathology”. In: *Medical Image Analysis* 83 (2023), p. 102655.
- [106] Parinaz Roshanzamir et al. “How inter-rater variability relates to aleatoric and epistemic uncertainty: a case study with deep learning-based paraspinal muscle segmentation”. In: *International Workshop on Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*. Springer. 2023, pp. 74–83.
- [107] Kumud Lakara and Matias Valdenegro-Toro. “Disentangled uncertainty and out of distribution detection in medical generative models”. In: *arXiv preprint arXiv:2211.06250* (2022).
- [108] Eoin Delaney, Derek Greene, and Mark T Keane. “Uncertainty estimation and out-of-distribution detection for counterfactual explanations: Pitfalls and solutions”. In: *arXiv preprint arXiv:2107.09734* (2021).
- [109] Jishnu Mukhoti et al. “Raising the bar on the evaluation of out-of-distribution detection”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 4365–4375.
- [110] Yucheng Tang et al. “Self-supervised pre-training of swin transformers for 3d medical image analysis”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 20730–20740.
- [111] Murat Sensoy, Lance Kaplan, and Melih Kandemir. “Evidential deep learning to quantify classification uncertainty”. In: *Advances in neural information processing systems* 31 (2018).
- [112] Yiding Jiang et al. “Methods and analysis of the first competition in predicting generalization of deep learning”. In: *NeurIPS 2020 Competition and Demonstration Track*. PMLR. 2021, pp. 170–190.

Chapter 9

Appendix

9.1 Overview previous work in uncertainty computation for medical image segmentation

Jungo et al. [79] applied MC-dropout in 2018 for brain tumor cavity segmentation and obtained a doubt score for one predicted binary image. A threshold value for the doubt score enabled flagging contours for manual validation. They evaluated the relation between the doubt-score and the dice coefficient between the clinical contour (CC) and the deep learning contour (DLC).

DeVries et al. [9] used a second network to predict quality of autosegmentation based on the input image and the uncertainty map obtained from the prior autosegmentation model. They compared MC-dropout with a logit variance estimation and the learned confidence estimate, which follows a similar approach as the logit variance estimation proposed by Kendall et al. Uncertainty prediction was applied to skin lesion segmentation and compared to the Jaccard index between the CC and DLC with a threshold for manual review of 0.7. However, they find similar quality between the metrics, but also compared to the maximum Softmax probability, which is the standard output of a DL model. Gal et al [34] argued that this measure is often misinterpreted as output uncertainty.

Nair et al. [71] presented multiple uncertainty estimates based on MC-dropout for categorisation of segmentation into lesion, non-lesion and uncertain in multiple sclerosis imaging. They compared logit output variance, variance of samples, predictive entropy and mutual information, as they all capture different types of uncertainty (variation in output labels caused by noise in input, reliability of the model and prediction made). All uncertainty measures improve overall autosegmentation error detection, but reflect different instances where the model was uncertain. The measurements can be combined to indicate areas where a model is certain that its prediction is uncertain.

Roy et al. [78] applied MC-dropout with entropy to whole-brain autosegmentation. They showed that uncertainty measures are highly correlated to the dice score between CC and DLC. As an uncertainty measure they used the intersection over overlap metric, which denotes the overlap between all MC samples.

Wang et al. [37] proposed a test-time augmentation-based aleatoric uncertainty and also applied MC-dropout for epistemic uncertainty. They evaluated these uncertainty metrics by assessing if the dice score between the and the average symmetric surface distance improved when the uncertainty areas were taken into account. They found that test-time augmentation was superior compared to MC-dropout.

Leousi et al.[42] applied MC-dropout, deep ensemble and logit variance estimation as proposed by Kendall et al. They compared uncertainty with accuracy, precision, DSC, Hausdorff distance and average surface distance. They observed that uncertainty can best be deployed for accuracy detection. Furthermore, they found that OOD data does not necessarily correlate with increased uncertainty.

Jungo et al.[79] continued their work from 2018 in a successive paper where they evaluated several uncertainty estimation methods for brain tumor segmentation in terms of calibration, segmentation error localisation and segmentation failure detection. They found that all methods (including Softmax probability of which Gal et al.[28] argued that it is not uncertainty) performed similarly, except for aleatoric uncertainty which performed worst. Furthermore, they argued that aggregation of uncertainties and/or radiomics is promising as voxel-wise uncertainty is difficult to interpret.

Van Rooij et al.[80] applied MC-dropout and MC-dropout with added gaussian noise to the model parameters for two head and neck cancer organs at risk . They provided each voxel with an uncertainty score which was thresholded to assess an autosegmentation voxel as true or false positive/negative as well as a whole structure-level uncertainty score. They observed more uncertainty in areas less consistently contoured and that uncertainty estimation improved by adding Gaussian noise.

Bhat et al.[98] compared the influence of several uncertainty methods (MC-dropout, deep ensemble, test-time augmentation) on liver lesion autosegmentation to shape-based features to classify autosegmentation quality (acceptable/inacceptable). They showed that uncertainty estimates do not contribute to reducing false positives, but shape-based features do.

Sahlsten et al.[81] applied MC-dropout and deep ensemble to GTV autosegmentation by obtaining the coefficient of variation, structure expected entropy, structure predictive entropy and structure mutual information, and evaluated it to qualify autosegmentation performance (described by the DSC, mean surface distance and hausdorff distance. They observed similar but distinct utility of the two uncertainty methods for predicting segmentation quality and referral performance.

Outeiral et al.[82] compared the autosegmentation NN output before binarization as a score map to autosegmentation quality (DSC, Hausdorff distance, mean surface dice and surface dice) in cervical cancer patients.

9.2 nnUNet network configuration and training parameter settings

This is the configuration used for nnUNet training (without mirroring enabled in data augmentation):

```
Configuration name: 3d_fullres
{
  "dataset_name": "Dataset001_HNC-B",
  "plans_name": "nnUNetPlans",
  "original_median_spacing_after_transp": [
    2.0,
    0.9765625,
    0.9765625
  ],
  "original_median_shape_after_transp": [
    175,
    512,
    512
  ],
  "image_reader_writer": "SimpleITKIO",
  "transpose_forward": [
    0,
    1,
    2
  ],
  "transpose_backward": [
    0,
    1,
    2
  ],
  "configurations": {
    "2d": {
      "data_identifier": "nnUNetPlans_2d",
      "preprocessor_name": "DefaultPreprocessor",
      "batch_size": 13,
      "patch_size": [
        512,
        448
      ],
      "median_image_size_in_voxels": [
        512.0,
        512.0
      ],
      "spacing": [
        0.9765625,
        0.9765625
      ],
      "normalization_schemes": [
        "CTNormalization"
      ],
      "use_mask_for_norm": [
        false
      ],
      "UNet_class_name": "PlainConvUNet",
      "UNet_base_num_features": 32,
      "n_conv_per_stage_encoder": [
        2,
        2,
        2,
        2,
        2,
        2,
        2
      ],
      "n_conv_per_stage_decoder": [
        2,
        2,
        2,
        2,
        2,
        2
      ]
    }
  }
}
```



```

    },
    "resampling_fn_seg_kwargs": {
        "is_seg": true,
        "order": 1,
        "order_z": 0,
        "force_separate_z": null
    },
    "resampling_fn_probabilities": "resample_data_or_seg_to_shape",
    "resampling_fn_probabilities_kwargs": {
        "is_seg": false,
        "order": 1,
        "order_z": 0,
        "force_separate_z": null
    },
    "batch_dice": true
},
"3d_lowres": {
    "data_identifier": "nnUNetPlans_3d_lowres",
    "preprocessor_name": "DefaultPreprocessor",
    "batch_size": 2,
    "patch_size": [
        64,
        192,
        192
    ],
    "median_image_size_in_voxels": [
        103,
        292,
        292
    ],
    "spacing": [
        3.404866122479809,
        1.712408254958107,
        1.712408254958107
    ],
    "normalization_schemes": [
        "CTNormalization"
    ],
    "use_mask_for_norm": [
        false
    ],
    "UNet_class_name": "PlainConvUNet",
    "UNet_base_num_features": 32,
    "n_conv_per_stage_encoder": [
        2,
        2,
        2,
        2,
        2,
        2
    ],
    "n_conv_per_stage_decoder": [
        2,
        2,
        2,
        2,
        2
    ],
    "num_pool_per_axis": [
        4,
        5,
        5
    ],
    "pool_op_kernel_sizes": [
        [
            1,
            1,
            1
        ],
        [
            2,
            2,
            2
        ]
    ]
}

```

```

    ],
    [
        2,
        2,
        2
    ],
    [
        2,
        2,
        2
    ],
    [
        2,
        2,
        2
    ],
    [
        1,
        2,
        2
    ]
],
"conv_kernel_sizes": [
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ]
],
"unet_max_num_features": 320,
"resampling_fn_data": "resample_data_or_seg_to_shape",
"resampling_fn_seg": "resample_data_or_seg_to_shape",
"resampling_fn_data_kwargs": {
    "is_seg": false,
    "order": 3,
    "order_z": 0,
    "force_separate_z": null
},
"resampling_fn_seg_kwargs": {
    "is_seg": true,
    "order": 1,
    "order_z": 0,
    "force_separate_z": null
},
"resampling_fn_probabilities": "resample_data_or_seg_to_shape",
"resampling_fn_probabilities_kwargs": {
    "is_seg": false,
    "order": 1,

```

```

        "order_z": 0,
        "force_separate_z": null
    },
    "batch_dice": false,
    "next_stage": "3d_cascade_fullres"
},
"3d_fullres": {
    "data_identifier": "nnUNetPlans_3d_fullres",
    "preprocessor_name": "DefaultPreprocessor",
    "batch_size": 2,
    "patch_size": [
        64,
        160,
        160
    ],
    "median_image_size_in_voxels": [
        175.0,
        512.0,
        512.0
    ],
    "spacing": [
        2.0,
        0.9765625,
        0.9765625
    ],
    "normalization_schemes": [
        "CTNormalization"
    ],
    "use_mask_for_norm": [
        false
    ],
    "UNet_class_name": "PlainConvUNet",
    "UNet_base_num_features": 32,
    "n_conv_per_stage_encoder": [
        2,
        2,
        2,
        2,
        2,
        2
    ],
    "n_conv_per_stage_decoder": [
        2,
        2,
        2,
        2,
        2
    ],
    "num_pool_per_axis": [
        4,
        5,
        5
    ],
    "pool_op_kernel_sizes": [
        [
            1,
            1,
            1
        ],
        [
            1,
            2,
            2
        ],
        [
            2,
            2,
            2
        ],
        [
            2,
            2,
            2
        ]
    ]
}

```

```

    ],
    [
        2,
        2,
        2
    ],
    [
        2,
        2,
        2
    ]
],
"conv_kernel_sizes": [
    [
        1,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ],
    [
        3,
        3,
        3
    ]
],
"unet_max_num_features": 320,
"resampling_fn_data": "resample_data_or_seg_to_shape",
"resampling_fn_seg": "resample_data_or_seg_to_shape",
"resampling_fn_data_kwargs": {
    "is_seg": false,
    "order": 3,
    "order_z": 0,
    "force_separate_z": null
},
"resampling_fn_seg_kwargs": {
    "is_seg": true,
    "order": 1,
    "order_z": 0,
    "force_separate_z": null
},
"resampling_fn_probabilities": "resample_data_or_seg_to_shape",
"resampling_fn_probabilities_kwargs": {
    "is_seg": false,
    "order": 1,
    "order_z": 0,
    "force_separate_z": null
},
"batch_dice": true
},
"3d_cascade_fullres": {
    "inherits_from": "3d_fullres",
    "previous_stage": "3d_lowres"
}
}

```

```

"experiment_planner_used": "ExperimentPlanner",
"label_manager": "LabelManager",
"foreground_intensity_properties_per_channel": {
  "0": {
    "max": 22704.0,
    "mean": 153.48446655273438,
    "median": 52.0,
    "min": -11652.0,
    "percentile_00_5": -302.0,
    "percentile_99_5": 1668.0,
    "std": 332.80194091796875
  }
}
}
}

```

9.3 nnUNet training commands

The commands put in the Hábrók high-performance computer cluster for **nnUNet!** (**nnUNet!**) training. Various options are provided: basic training, training for 4000 epochs (instead of 1000), training without mirroring during data augmentation and training for a deep ensemble model.

```

1 ## Y800
2 ##SBATCH --time=07:00:00
3 #
4 ##SBATCH --partition=cpu
5 ##SBATCH --cpus-per-task=100
6 ##SBATCH --mem=128GB # (the nodes have 128GB, but a bit is reserved to the operating system)
7 ##SBATCH --cpus-per-task=5
8 #
9 ##A100
10 ##SBATCH --time=24:00:00
11 #
12 ##SBATCH --partition=cpu
13 ##SBATCH --cpus-per-task=100
14 ##SBATCH --mem=128GB # (the nodes have 128GB, but a bit is reserved to the operating system)
15 ##SBATCH --cpus-per-task=5
16 #
17 module purge
18 module load Python/3.9.6-GCCcore-11.2.0
19 #
20 python3 --version
21 #
22 pip install --upgrade pip
23 pip install --upgrade wheel
24 #
25 source /home4/p316227/venvs/nnUnet/bin/activate
26 #
27 cd /scratch/p316227/nnUnet_folder/
28 #
29 export nnUnet_raw="/scratch/p316227/nnUnet_folder/nnUnetFrame/Dataset/nnUnet_raw"
30 export nnUnet_results="/scratch/p316227/nnUnet_folder/nnUnetFrame/Dataset/nnUnet_results"
31 export nnUnet_preprocessed="/scratch/p316227/nnUnet_folder/nnUnetFrame/Dataset/nnUnet_preprocessed"
32 export nnUnet_n_proc_DA=28
33 #
34 ##### data preprocessing (making data fingerprint)
35 nnUnetv2_plan_and_preprocess -d 1 --verify_dataset_integrity
36 #
37 ##### original nnUnet training
38 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 0 --c
39 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 1 --c
40 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 2 --c
41 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 3 --c
42 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 4 --c
43 #
44 ##### elongated nnUnet training (4000 epochs)
45 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 0 -tr nnUnetTrainer_4000epochs --c
46 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 1 -tr nnUnetTrainer_4000epochs --c
47 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 2 -tr nnUnetTrainer_4000epochs --c
48 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 3 -tr nnUnetTrainer_4000epochs --c
49 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 4 -tr nnUnetTrainer_4000epochs --c
50 #
51 ##### no mirroring training
52 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 0 -tr nnUnetTrainerNoMirroring --c
53 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 1 -tr nnUnetTrainerNoMirroring --c
54 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 2 -tr nnUnetTrainerNoMirroring --c
55 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 3 -tr nnUnetTrainerNoMirroring --c
56 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres 4 -tr nnUnetTrainerNoMirroring --c
57 #
58 ##no folds --> For deep ensemble modelling
59 OMP_NUM_THREADS=1 nnUnetv2_train 001 3d_fullres -F all -tr nnUnetTrainerNoMirroring --c

```

Figure 9.1: nnUnet commands in Hábrók for preprocessing and training

9.4 Patient level autosegmentation analysis

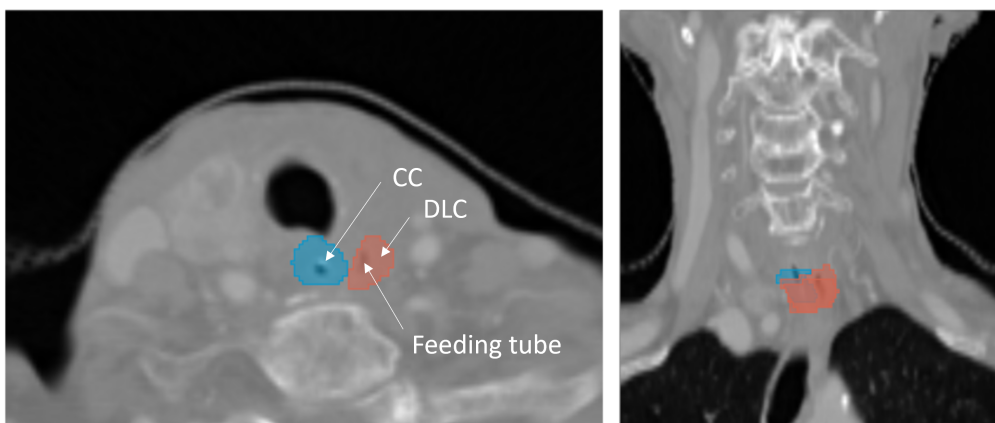


Figure 9.2: Oesophagus: feeding tube leading to low geometric performance autosegmentation

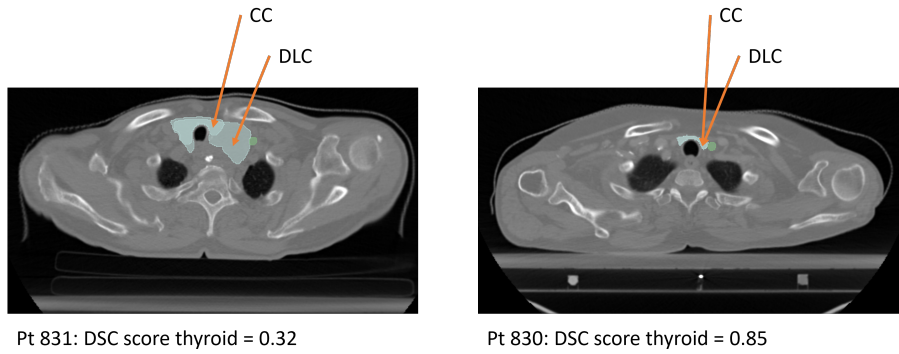


Figure 9.3: Thyroid: left carotid artery guiding the thyroid autosegmentation contour

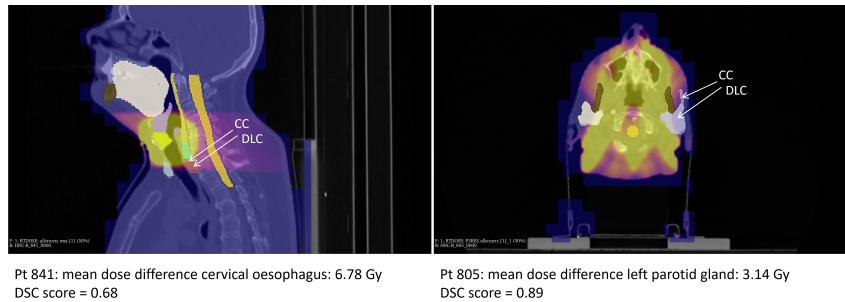


Figure 9.4: Oesophagus: cranial and caudal extension in high dose area. Parotid anterior extension in high dose area

9.5 DIP vs DIE vs DIA

OAR contouring inaccuracies seep through various parts of the clinical RT workflow. Firstly, contouring determines the RT treatment plan, as with the treatment plan, an optimal distribution is pursued of minimal dose on the OARs and maximum coverage of the target. Thus, the differences between a DL contour and CC have an effect on the treatment plan (DIP) Secondly, the quality of the treatment plan is evaluated on the OARs by looking at the mean and max dose constraints for example and thereby minimise RT related toxicities. This information is not only used to evaluate the treatment plan but also used for model-based patient selection to decide if patients receive photon or proton therapy. The differences between a DL contour and CC thus also have an effect on the evaluation of a treatment plan (DIE).

The DIE is computed by comparing the mean dose of a DL contour and CC on the treatment plan based on the DL contour set. The DIP is computed by comparing the dose value of a DL contour between the treatment plan based on the DL contour set and the treatment plan based on the CC set. For computing the DIP and DIE new treatment plans thus have to be computed for the DL contour set.

Previous research has shown that the DIE can be approximated by evaluating both the DL contour and CC on the treatment plan based on the CC set. This alleviates time and resources, as no new treatment plans have to be made. As the previous research by van Dijk et al. [6] used the treatment plans based on the CC set, i.e. the approximated DIE, and the goal is to obtain similar autosegmentation performance, this metric will be computed for the new model as well and other dosimetric measures will be neglected.

9.6 Extra geometric performance metrics nnUNet

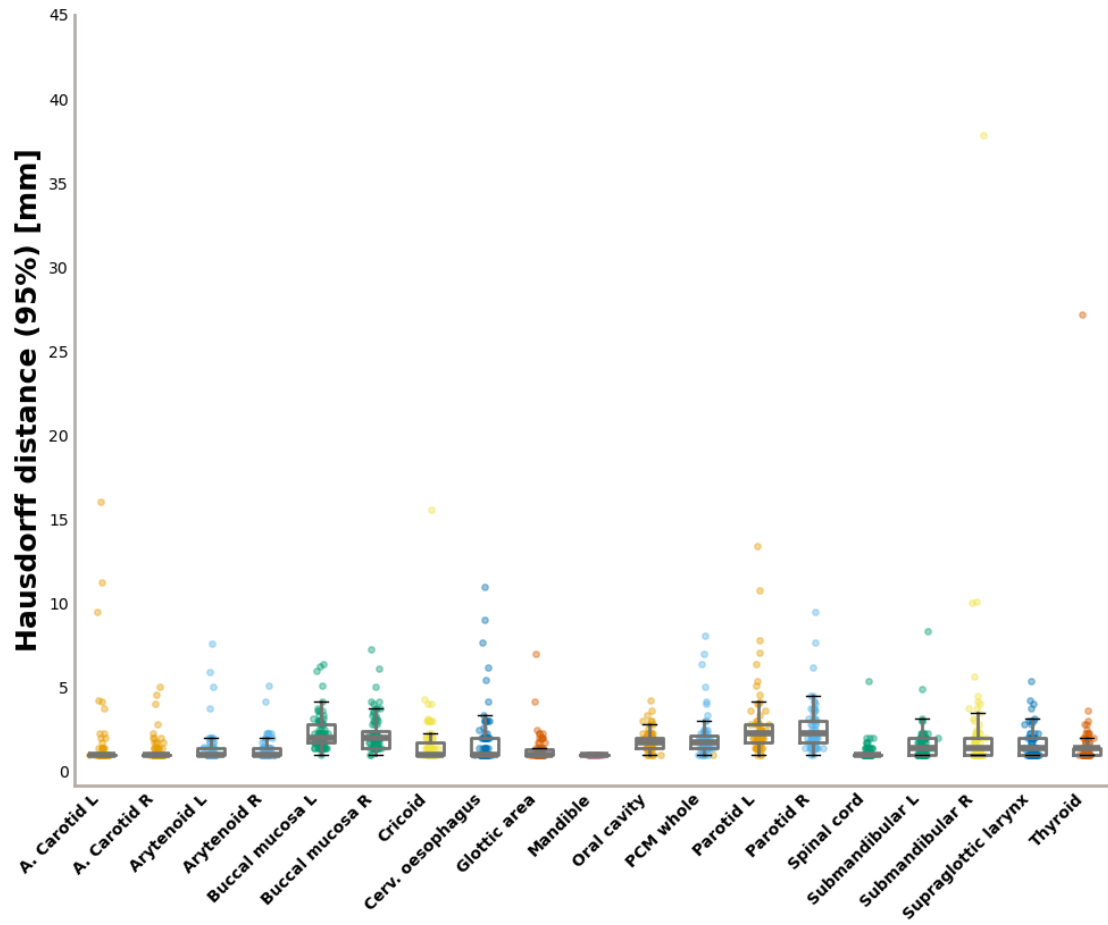


Figure 9.5: Hausdorff distance 95th percentile

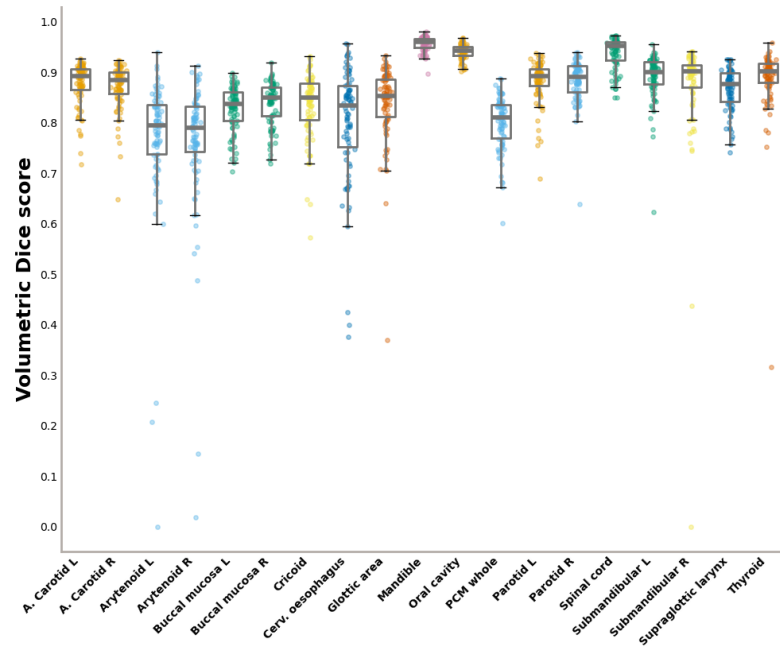


Figure 9.6: Surface DSC score

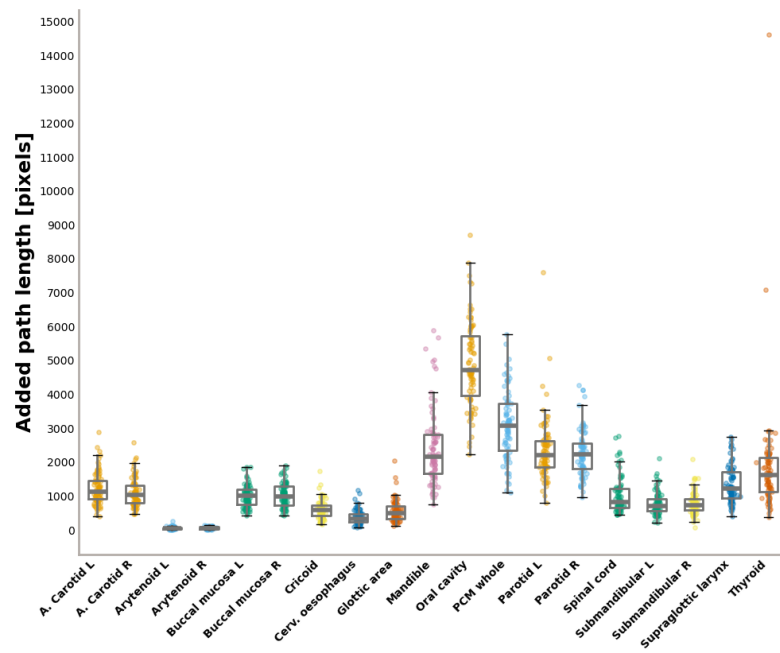


Figure 9.7: Added path length

9.7 Extra dosimetric performance metric nnUNet

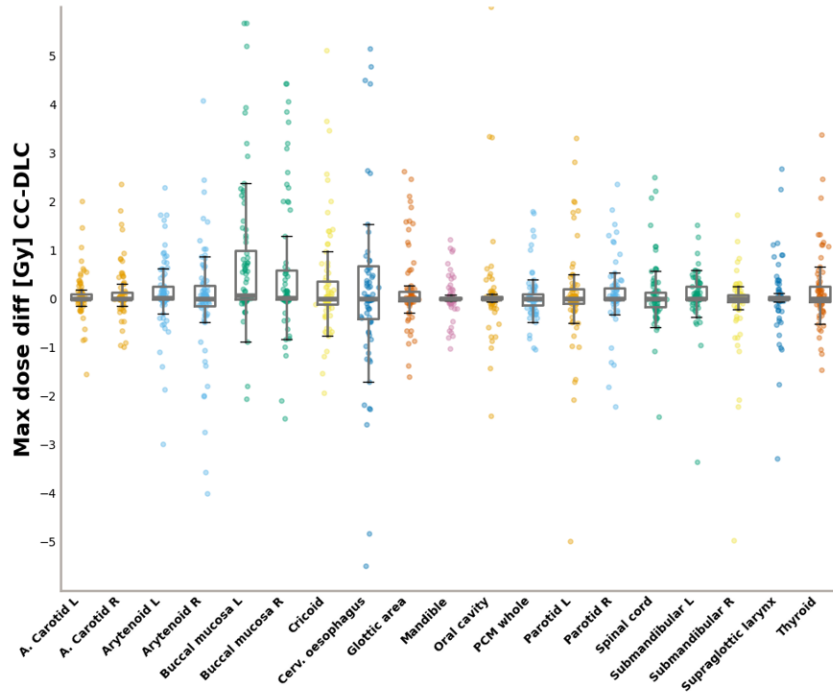


Figure 9.8: Box plots of max dose difference per OAR for all test patients on nnUNet autosegmentation model

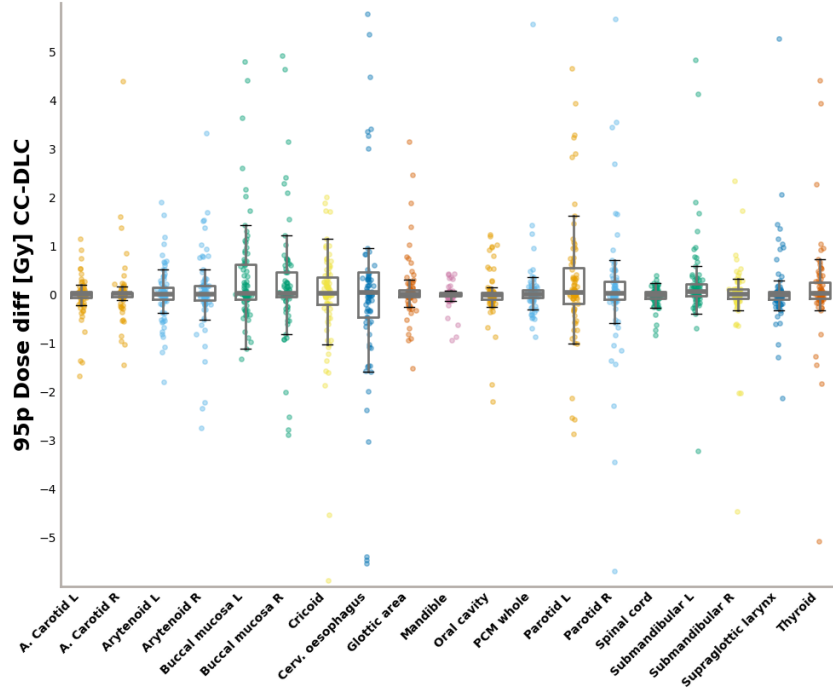


Figure 9.9: Dose received in 95% of the volume (D95)

9.8 nnUNet with and without mirroring

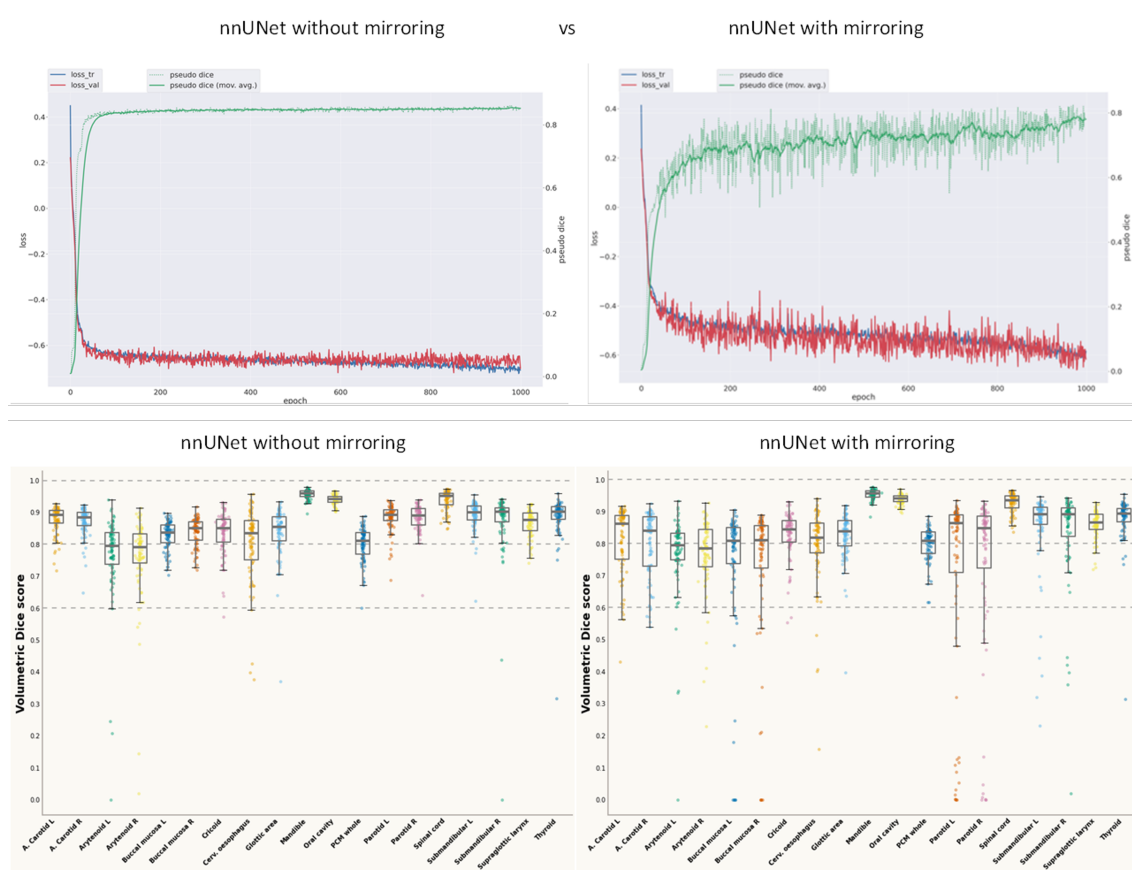


Figure 9.10: nnUNet with and without mirroring. Training loss and validation DSC scores and geometric performance on test set with volumetric DSC scores.

9.9 Wilcoxon signed rank test for geometric and dosimetric differences

Table 9.1: Wilcoxon Signed Rank test for comparison of differences between autosegmentation output between networks with and without added uncertainty computation. Geometric evaluation

Uncertainty method	Dropout location	Dropout rate	Volumetric DSC	HD	HD95	APL	Surface DSC
MC Dropout	All	0.1	Z=-5.995 p < 0.001	Z=-4.980 p < 0.001	Z=-3.324 p < 0.001	Z=-5.004 p < 0.001	Z=-6.719 p < 0.001
MC Dropout	All	0.2	Z=-9.348 p < 0.001	Z=-7.481 p < 0.001	Z=-6.784 p < 0.001	Z=-8.876 p < 0.001	Z=-3.229 p < 0.001
MC Dropout	All	0.3	Z=-11.099 p < 0.001	Z=-10.146 p < 0.001	Z=-10.197 p < 0.001	Z=-11.005 p < 0.001	Z=-3.900 p < 0.001
MC Dropout	All	0.4	Z=-8.979 p < 0.001	Z=-8.979 p < 0.001	Z=-8.979 p < 0.001	Z=-8.979 p < 0.001	Z=-4.025 p < 0.001
MC Dropout	All	0.5	NaN	NaN	NaN	NaN	NaN
MC Dropout	Central	0.1	Z=-0.062 p=0.951	Z=-2.261 p=0.024	Z=-0.982 p=0.326	Z=-0.304 p=0.761	Z=-5.946 p < 0.001
MC Dropout	Central	0.2	Z=-2.346 p=0.019	Z=-2.402 p=0.016	Z=-1.940 p=0.052	Z=-2.116 p=0.034	Z=-5.091 b
MC Dropout	Central	0.3	Z=-3.700 p < 0.001	Z=-4.139 p < 0.001	Z=-2.968 p=0.003	Z=-2.526 p=0.012	Z=-5.395 p < 0.001
MC Dropout	Central	0.4	Z=-5.247 p < 0.001	Z=-5.397 p < 0.001	Z=-4.413 p < 0.001	Z=-4.490 p < 0.001	Z=-4.244 p < 0.001
MC Dropout	Central	0.5	Z=-7.536 p < 0.001	Z=-6.865 p < 0.001	Z=-5.754 p < 0.001	Z=6.999 p < 0.001	Z=-2.617 p=0.009
Deep Ensemble	-	-	Z=-2.784 p=0.005	Z=-0.236 p=0.813	Z=-0.408 p=0.683	Z=-6.844 p < 0.001	Z=-1.620 p=0.105
TTA	-	-	Z=-2.195 p=0.028	Z=-0.192 p=0.848	Z=-1.689 p=0.091	Z=-0.872 p=0.383	Z=-2.897 p=0.00

MC = Monte Carlo, TTA = Test Time Augmentation, Z = Z-score Wilcoxon rank-sum, p = p-value.
A p-value above 0.05 indicates a statistically significant difference between the networks with and without added uncertainty computation. Shown in bold.

Table 9.2: Wilcoxon Signed Rank test for comparison of differences between autosegmentation output between networks with and without added uncertainty computation. Dosimetric evaluation

Uncertainty method	Dropout location	Dropout rate	Mean dose difference CC-DLC	Max dose difference CC-DLC	ABS Mean dose difference ABS(CC-DLC)	ABS Max dose difference ABS(CC-DLC)
MC Dropout	All	0.1	Z=-3.899 p < 0.001	Z=-7.556 p < 0.001	Z=-2.346 p=0.019	Z=-2.161 p=0.031
MC Dropout	All	0.2	Z=-4.271 p < 0.001	Z=-9.664 p < 0.001	Z=-4.209 p < 0.001	Z=-3.723 p < 0.001
MC Dropout	All	0.3	Z=-3.621 p < 0.001	Z=-10.869 p < 0.001	Z=-6.557 p < 0.001	Z=-7.630 p < 0.001
MC Dropout	All	0.4	Z=-2.525 p < 0.001	Z=-8.937 p < 0.001	Z=-8.014 p < 0.001	Z=-8.879 p < 0.001
MC Dropout	All	0.5	NaN	NaN	NaN	NaN
MC Dropout	Central	0.1	Z=-0.59 p=0.953	Z=-1.625 p=0.104	Z=-1.700 p=0.089	Z=-1.942 p=0.052
MC Dropout	Central	0.2	Z=-0.125 p=0.901	Z=-2.619 p=0.009	Z=-1.982 p=0.047	Z=-2.369 p = 0.018
MC Dropout	Central	0.3	Z=-0.060 p=0.952	Z=-2.936 p=0.003	Z=-1.823 p=0.068	Z=-1.837 p = 0.066
MC Dropout	Central	0.4	Z=-0.258 p=0.796	Z=-4.581 p < 0.001	Z=-3.178 p < 0.001	Z=-1.962 p = 0.050
MC Dropout	Central	0.5	Z=-1.009 p=0.313	Z=-5.503 p < 0.001	Z=-4.142 p < 0.001	Z=-2.588 p=0.10
Deep Ensemble	-	-	Z=-1.555 p=0.120	Z=-11.341 p < 0.001	Z=-0.599 p=0.549	Z=-0.898 p=0.369
Deep Ensemble	-	-	Z=-1.804 p=0.071	Z=-2.897 p=0.004	Z=-2.718 p=0.007	Z=-2.724 p=0.006

MC = Monte Carlo, TTA = Test Time Augmentation, CC = Clinical Contour, DLC = Deep Learning Contour, ABS = absolute Z = Z-score Wilcoxon rank-sum, p = p-value.

A p-value above 0.05 indicates a statistically significant difference between the networks with and without added uncertainty computation. Shown in bold.

9.10 Error localisation

9.10.1 Utilising uncertainty to locate errors

A possible future for a practical application of uncertainty is to guide autosegmentation assessors, radiotherapy technologist (RTT)s, to inaccuracies which can decrease intra-observer variability and accelerate autosegmentation assessment. [33] Keeping this in mind, it is therefore essential that areas of uncertainty align with areas of inaccuracy. Similarly, areas of certainty should align with areas of accuracy. An evaluation standpoint for uncertainty is thus that areas of uncertainty align with areas of error.

Uncertainty-error overlap

Jungo et al. [79] proposed the uncertainty error overlap (U-E) which measures the Dice coefficient between the regions where the model is uncertain U and the segmentation error E which is defined as the area of FPs and FNs:

$$U - E = \frac{2|U \cap E|}{|U| + |E|} \quad (9.1)$$

The volume of uncertain voxels is defined by a threshold on the uncertainty values. The threshold, τ is often determined independently per experiment by evaluating a range of thresholds from 0.05 to 0.95 in steps of 0.05.

9.10.2 Error Localisation

A quantitative evaluation of error and uncertainty overlap is provided in Table 9.3 and 9.4 for multiclass and classwise uncertainty computation metrics respectively. Uncertainty-error overlap scores for entropy are highest compared to variance and mutual information. For MC-dropout applied on the central resolutions with a dropout rate of 0.1, U-E overlap scores are 0.45, 0.27 and 0.30 respectively for entropy, variance and mutual information. MC-dropout applied on the all resolutions leads to U-E overlap scores of 0.46, 0.39 and 0.40. For Deep Ensemble modelling and Test-Time augmentation, U-E overlap scores deviate at most 0.01 points from the scores observed with MC-dropout applied on all resolutions. U-E overlap scores decrease when decrease the dropout rate from 0.1 to 0.3, however increase again when applying a dropout rate of 0.4 and 0.5. However, when looking at the accuracy map, for those dropout rates the autosegmentation output vanishes. Classwise uncertainty metrics are minimally different, where classwise entropy results in U-E of 0.46 or 0.45 for all uncertainty methods and classwise variance results in U-E values of 0.39 or 0.40 for all uncertainty methods when neglecting false MC-dropout results.

The quantitative error localisation analysis with the U-E value leads to comparable results as obtained by Jungo et al. [79] where values around 0.43 were observed. They did not report their uncertainty threshold values, but for the results of our analysis, thresholds for all uncertainty metrics of 0.20, 0.05 and 0.05 performed best for entropy, variance and mutual information respectively. A remarkable observation here is the consistency between each uncertainty metric per uncertainty method. For all uncertainty methods, every uncertainty metric has the same optimal threshold (when only considering the best-calibrated uncertainty maps for MC-dropout). However, this metric might be too coarse to investigate the visual observation that with variance and mutual information, uncertainty seemed to be more located at specific areas in an uncertainty map.

Table 9.3: Calibration metrics per uncertainty method, hyperparameter tuning and metric

Uncertainty method	Dropout location	Dropout rate	Metric	Best U-E threshold	U-E score
MC Dropout	All	0.1	Multiclass Entropy	0.20	0.46±0.01
MC Dropout	All	0.2	Multiclass Entropy	0.35	0.44±0.01
MC Dropout	All	0.3	Multiclass Entropy	0.05	0.40±0.03
MC Dropout	All	0.4	Multiclass Entropy	0.05	0.82±0.03
MC Dropout	All	0.5	Multiclass Entropy	0.05	0.71±0.08
MC Dropout	Central	0.1	Multiclass Entropy	0.20	0.45±0.01
MC Dropout	Central	0.2	Multiclass Entropy	0.20	0.45±0.01
MC Dropout	Central	0.3	Multiclass Entropy	0.20	0.46±0.01
MC Dropout	Central	0.4	Multiclass Entropy	0.25	0.45±0.02
MC Dropout	Central	0.5	Multiclass Entropy	0.30	0.44±0.03
Deep Ensemble	-	-	Multiclass Entropy	0.20	0.46±0.02
TTA	-	-	Multiclass Entropy	0.20	0.46±0.01
MC Dropout	All	0.1	Multiclass Variance	0.05	0.39±0.06
MC Dropout	All	0.2	Multiclass Variance	0.05	0.41±0.02
MC Dropout	All	0.3	Multiclass Variance	0.05	0.38±0.03
MC Dropout	All	0.4	Multiclass Variance	0.05	0.71±0.04
MC Dropout	All	0.5	Multiclass Variance	0.05	0.32±0.17
MC Dropout	Central	0.1	Multiclass Variance	0.05	0.27±0.04
MC Dropout	Central	0.2	Multiclass Variance	0.05	0.32±0.02
MC Dropout	Central	0.3	Multiclass Variance	0.05	0.37±0.02
MC Dropout	Central	0.4	Multiclass Variance	0.05	0.39±0.02
MC Dropout	Central	0.5	Multiclass Variance	0.05	0.39±0.03
Deep Ensemble	-	-	Multiclass Variance	0.05	0.40±0.01
TTA	-	-	Multiclass Variance	0.05	0.40±0.01
MC Dropout	All	0.1	Mutual information	0.05	0.40±0.05
MC Dropout	All	0.2	Mutual information	0.10	0.38±0.02
MC Dropout	All	0.3	Mutual information	0.05	0.39±0.03
MC Dropout	All	0.4	Mutual information	0.05	0.78±0.03
MC Dropout	All	0.5	Mutual information	0.05	0.53±0.13
MC Dropout	Central	0.1	Mutual information	0.05	0.30±0.05
MC Dropout	Central	0.2	Mutual information	0.05	0.35±0.02
MC Dropout	Central	0.3	Mutual information	0.05	0.38±0.03
MC Dropout	Central	0.4	Mutual information	0.05	0.39±0.03
MC Dropout	Central	0.5	Mutual information	0.05	0.38±0.03
Deep Ensemble	-	-	Mutual information	0.05	0.39±0.01
TTA	-	-	Mutual information	0.05	0.40±0.02

MC = Monte Carlo, TTA = Test Time Augmentation, CC = Clinical Contour, DLC = Deep Learning Contour.

*Results are averaged over all ROIs uncertainty method and uncertainty metric, and also, when applicable per dropout location and per dropout rate.

Table 9.4: Error localisation metrics per uncertainty method, hyperparameter tuning and metric

Uncertainty method	Dropout location	Dropout rate	Metric	Best U-E threshold	U-E score
MC Dropout	All	0.1	Classwise Entropy	0.30	0.43±0.05
MC Dropout	All	0.2	Classwise Entropy	0.50	0.42±0.06
MC Dropout	All	0.3	Classwise Entropy	0.05	0.45±0.12
MC Dropout	All	0.4	Classwise Entropy	0.05	0.81±0.08
MC Dropout	All	0.5	Classwise Entropy	0.05	0.70±0.12
MC Dropout	Central	0.1	Classwise Entropy	0.20	0.43±0.05
MC Dropout	Central	0.2	Classwise Entropy	0.25	0.43±0.05
MC Dropout	Central	0.3	Classwise Entropy	0.25	0.43±0.05
MC Dropout	Central	0.4	Classwise Entropy	0.30	0.43±0.06
MC Dropout	Central	0.5	Classwise Entropy	0.40	0.42±0.06
TTA	-	-	Classwise Entropy	0.30	0.44±0.05
MC Dropout	All	0.1	Classwise Variance	0.05	0.41 ±0.07
MC Dropout	All	0.2	Classwise Variance	0.10	0.40 ±0.06
MC Dropout	All	0.3	Classwise Variance	0.05	0.41 ±0.10
MC Dropout	All	0.4	Classwise Variance	0.05	0.71 ±0.10
MC Dropout	All	0.5	Classwise Variance	0.05	0.32 ±0.22
MC Dropout	Central	0.1	Classwise Variance	0.05	0.36 ±0.07
MC Dropout	Central	0.2	Classwise Variance	0.05	0.38 ±0.06
MC Dropout	Central	0.3	Classwise Variance	0.05	0.40 ±0.06
MC Dropout	Central	0.4	Classwise Variance	0.05	0.40 ±0.06
MC Dropout	Central	0.5	Classwise Variance	0.05	0.40 ±0.07
Deep Ensemble	-	-	Classwise Variance	0.05	0.41±0.05
TTA	-	-	Classwise Variance	0.05	0.29±0.11

MC = Monte Carlo, TTA = Test Time Augmentation, CC = Clinical Contour, DLC = Deep Learning Contour.

Results are averaged over all ROIs uncertainty method and uncertainty metric, and also, when applicable per dropout location and per dropout rate.

9.11 OOD analysis: uncertainty map metrics

	Metal				Bolus				Bitelock			
	Baseline		No-metal model		Baseline		No-bolus model		Baseline		No-bitelock model	
	DE	MCD	DE	MCD	DE	MCD	DE	MCD	DE	MCD	DE	MCD
Mean uncertainty	0.65	0.73	0.65	0.73	0.75	0.84	0.75	0.64	0.88	0.92	1.00	0.94
Thresholded mean uncertainty												
T = 0.01	0.28	0.28	0.67	0.73	0.42	0.50	0.33	0.31	0.56	0.60	0.61	0.44
T = 0.05	0.25	0.29	0.66	0.71	0.44	0.56	0.40	0.34	0.54	0.70	0.53	0.53
T = 0.1	0.24	0.30	0.60	0.76	0.49	0.61	0.41	0.34	0.52	0.70	0.47	0.56
T = 0.3	0.32	0.38	0.63	0.60	0.60	0.68	0.47	0.29	0.54	0.74	0.56	0.50
T = 0.5	0.48	0.53	0.63	0.57	0.49	0.69	0.32	0.29	0.44	0.74	0.36	0.56
std uncertainty	0.54	0.70	0.71	0.76	0.76	0.83	0.70	0.57	0.88	0.92	0.92	0.81
Thresholded std uncertainty												
T = 0.01	0.25	0.30	0.69	0.76	0.50	0.62	0.43	0.34	0.54	0.66	0.53	0.56
T = 0.05	0.33	0.36	0.58	0.72	0.60	0.66	0.43	0.28	0.54	0.74	0.53	0.50
T = 0.1	0.36	0.39	0.56	0.61	0.62	0.67	0.47	0.28	0.52	0.74	0.50	0.50
T = 0.3	0.35	0.40	0.56	0.48	0.61	0.68	0.44	0.29	0.56	0.74	0.53	0.50
T = 0.5	0.61	0.68	0.37	0.48	0.44	0.68	0.33	0.33	0.40	0.70	0.44	0.56
Voxels above threshold												
T = 0.01	0.89	0.90	0.78	0.82	0.61	0.62	0.63	0.45	0.90	0.88	0.94	0.94
T = 0.1	0.87	0.88	0.78	0.86	0.59	0.58	0.57	0.42	0.90	0.94	0.97	0.94
T = 0.3	0.83	0.82	0.80	0.89	0.57	0.59	0.57	0.41	0.88	0.94	0.97	0.94
T = 0.5	0.50	0.57	0.80	0.81	0.60	0.65	0.54	0.39	0.72	1.00	0.75	0.72

Figure 9.11: Uncertainty histogram metrics on in-distribution and out-of-distribution data on out-of-distribution models and the baseline model.