Visual Analysis of Contour Ensembles in the Context of Radiotherapy

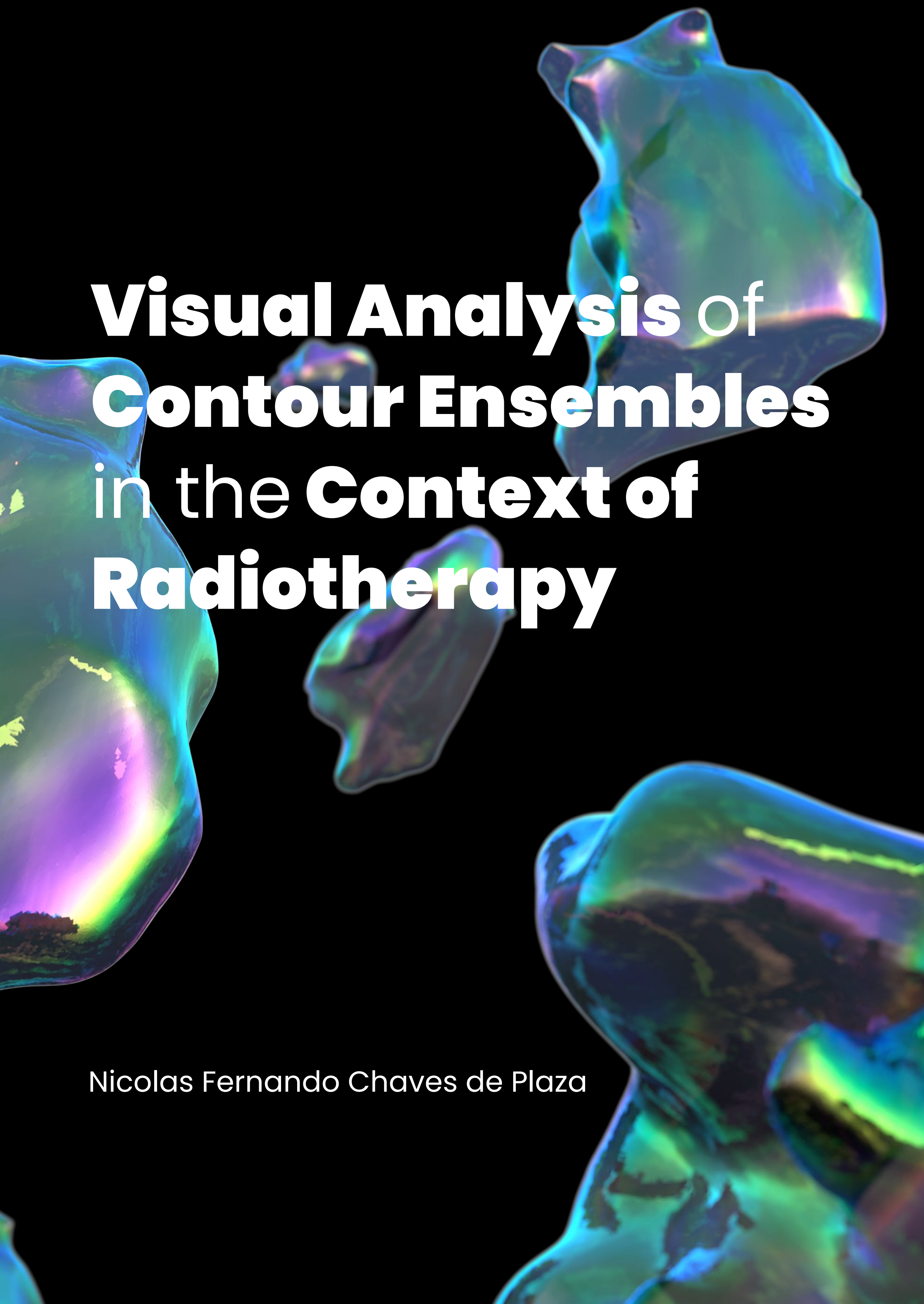
Chaves-de-Plaza, Nicolas F.

DOI
10.4233/uuid:2619c382-9b89-4eac-8fa3-5b8454973e5b

Publication date
2025

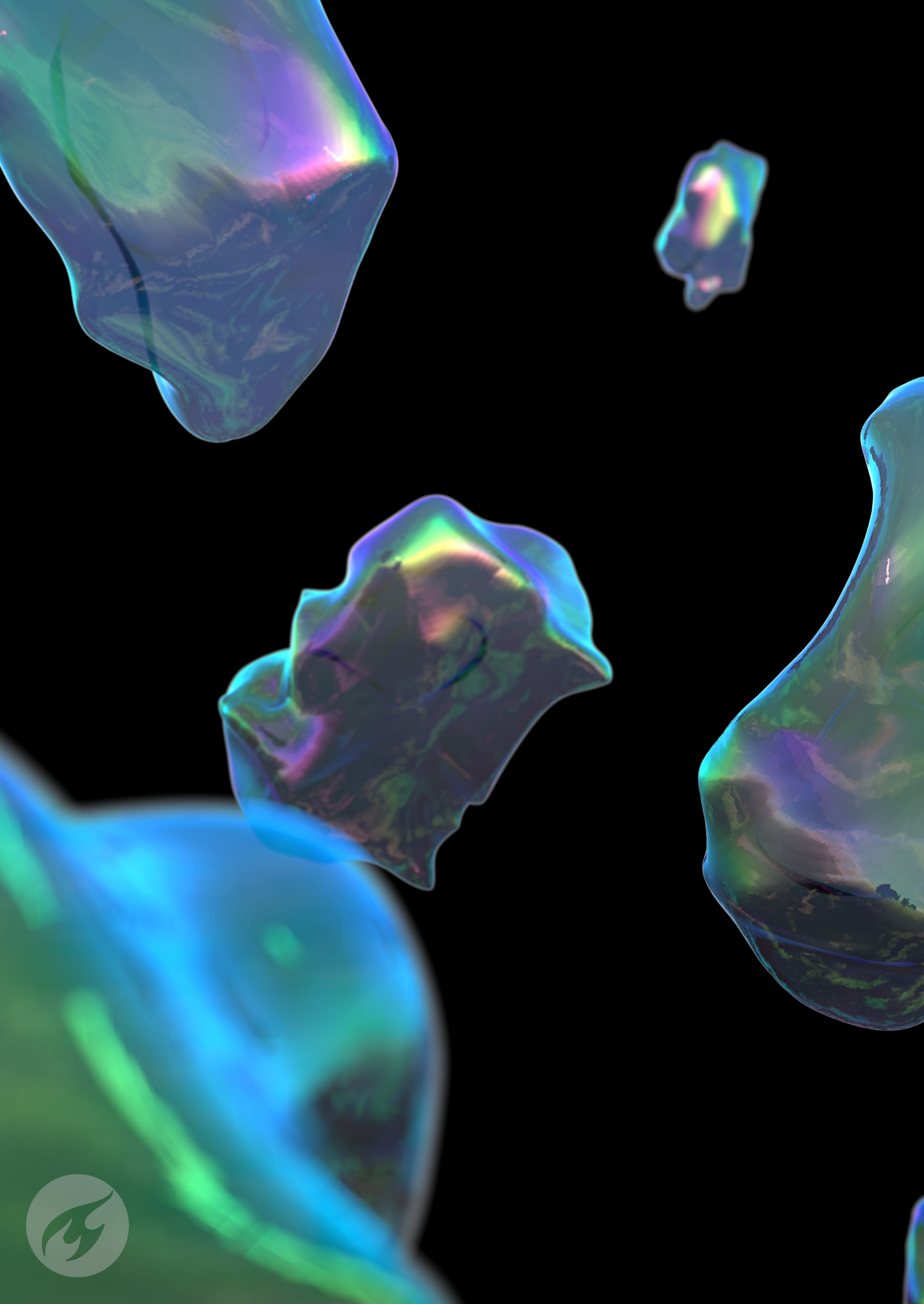
Document Version
Final published version

Citation (APA)
Chaves-de-Plaza, N. F. (2025). *Visual Analysis of Contour Ensembles in the Context of Radiotherapy*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:2619c382-9b89-4eac-8fa3-5b8454973e5b



Visual Analysis of **Contour Ensembles** in the **Context of Radiotherapy**

Nicolas Fernando Chaves de Plaza



Visual Analysis of Contour Ensembles in the Context of Radiotherapy

Visual Analysis of Contour Ensembles in the Context of Radiotherapy

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus, prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Wednesday 18 June 2025 at 10:00 o'clock

by

Nicolas Fernando CHAVES DE PLAZA

Master of Science in Computer Science,
Delft University of Technology, Netherlands

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Dr. K.A. Hildebrandt,	Delft University of Technology, *promotor*
Dr. R. van Egmond,	Delft University of Technology, *promotor*
Prof. dr. A. Vilanova Bartroli,	Delft University of Technology / Eindhoven University of Technology, *promotor*
Independent members:	
Prof. dr. ir. M.S. Kleinsmann,	Delft University of Technology
Prof. dr. R. Westermann,	Technical University of Munich, Germany
Prof. dr. K. Lawonn,	Friedrich Schiller University Jena, Germany
Prof. dr. N. Smit,	University of Bergen, Norway

Prof. dr. H. de Ridder of Delft University of Technology has contributed greatly to the preparation of this dissertation.

Keywords: Adaptive Radiotherapy, Medical Image Segmentation, Ensemble Methods, Quality Assessment, Uncertainty Visualization, Contour Analysis

Printed by: ProefschriftMaken

Cover by: Nicolas F. Chaves de Plaza

ISBN 978-94-6510-682-3

An electronic copy of this dissertation is available at https://repository.tudelft.nl/.

You can’t connect the dots looking forward;
you can only connect them looking backwards.
So you have to trust that the dots will
somehow connect in your future.

Steve Jobs

CONTENTS

Summary **xi**

Samenvatting **xv**

1. Introduction **1**

1.1. Clinical Context: Adaptive Radiotherapy 2
1.2. Medical Image Segmentation 4
1.3. AI-Supported Segmentation 5
1.4. Deep Ensemble AI 7
1.5. Thesis Outline 9
1.5.1. Towards Fast Human-Centred Contouring Workflows for Adaptive External Beam Radiotherapy 10
1.5.2. Implementation of Delineation Error Detection Systems in Time-Critical Radiotherapy: Do AI-Supported Optimization and Human Preferences Meet? 10
1.5.3. Inclusion Depth for Contour Ensembles 11
1.5.4. Depth for Multi-Modal Contour Ensembles 12
1.5.5. LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles 13

2. Towards Fast Human-Centered Contouring Workflows for Adaptive External Beam Radiotherapy **19**

2.1. Introduction 20
2.2. The Contouring Activity 22
2.3. Study of the Contouring Workflow 23
2.3.1. Study design 23
2.3.2. Data Analysis 24
2.4. Initial Contouring 25
2.4.1. Results 25
2.4.2. Discussion 28
2.5. Adaptive Contouring 28
2.5.1. Results 28
2.5.2. Discussion 31
2.6. Discussion 32
2.6.1. Inspection and Navigation 33
2.6.2. Editing 34
2.7. Limitations and Future Work 35
2.8. Conclusion 35

3. Implementation of Delineation Error Detection Systems in Time-Critical Radiotherapy: Do AI-Supported Optimization and Human Preferences Meet? 43
3.1. Introduction . . . 45
3.2. Related Work . . . 46
3.3. Materials and Methods . . . 47
3.3.1. Imaging Data . . . 47
3.3.2. AI Delineations, Uncertainty and Error . . . 49
3.3.3. Per-Slice Scores . . . 50
3.4. User Study: Workflow Comparison . . . 50
3.4.1. Study Setup . . . 51
3.4.2. Part 1: Non-Assisted Workflow . . . 53
3.4.3. Part 2: DEDS-Assisted Workflows . . . 56
3.4.4. Discussion . . . 58
3.5. Simulation Study: Assessing DEDS-Induced Time Gains . . . 59
3.5.1. Simulation Setup . . . 59
3.5.2. Results and Discussion . . . 61
3.6. Discussion . . . 63
3.7. Conclusion . . . 64

4. Inclusion Depth for Contour Ensembles 73
4.1. Introduction . . . 74
4.2. Related Work . . . 76
4.3. Background: Contour Depth and Boxplots . . . 77
4.3.1. Statistical Depth . . . 77
4.3.2. Contour Band Depth . . . 78
4.3.3. Contour Boxplots . . . 79
4.4. Inclusion Depth . . . 79
4.5. Epsilon Inclusion Depth . . . 81
4.6. Experiments . . . 83
4.6.1. Experimental Setup . . . 84
4.6.2. Experiment 1: Scaling Behavior . . . 86
4.6.3. Experiment 2: Outlier Detection . . . 86
4.6.4. Experiment 3: Estimator's Robustness . . . 89
4.7. Visual Comparison on Real Data . . . 90
4.7.1. Medical Image Segmentation Ensembles . . . 91
4.7.2. Meteorological Forecasting . . . 94
4.8. Discussion and Conclusion . . . 95

5. Depth for Multi-Modal Contour Ensembles 101
5.1. Introduction . . . 102
5.2. Related Work . . . 104
5.3. Background: Contour Depth . . . 105
5.4. Linear Epsilon Inclusion Depth Computation . . . 107
5.5. Fast Depth Recomputation . . . 108

5.6. Multi-Modal Analysis 110
5.6.1. Relative Depth 110
5.6.2. CDclust 112
5.6.3. CDclust Complexity 114
5.7. Experiments on Synthetic Data 114
5.7.1. Fast Computation of Contour Depth 114
5.7.2. Multi-Modal Contour Analysis 116
5.8. Case Studies 119
5.8.1. Segmentation Ensembles 121
5.8.2. Weather Forecast Ensembles 122
5.9. Discussion and Conclusions 123

6. LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles 129
6.1. Introduction 130
6.2. Related Work 131
6.3. Background 134
6.3.1. Pivot Algorithm 135
6.3.2. CN-Pivot Algorithm 135
6.4. Local-to-Global Correlation Clustering 136
6.4.1. The Local Step 136
6.4.2. The Global Step 136
6.4.3. Properties and Thresholds 140
6.4.4. Runtime Analysis and Exploration of ρ_t 141
6.5. Experiments and Results 142
6.5.1. Data 142
6.5.2. Setup 144
6.5.3. Experiment 1: Scaling 144
6.5.4. Experiment 2: Clustering Quality 146
6.5.5. Experiment 3: Correlation Threshold Exploration 148
6.5.6. Experiment 4: LoGCC’s Sensitivity to Decreasing ρ_l . 150
6.6. Discussion and Conclusion 151

7. Applying Contour Depth Analysis to Real-World Radiotherapy Challenges 159
7.1. Introduction 160
7.2. Methods and Materials 161
7.2.1. Data 161
7.2.2. Consensus Extraction and Similarity Metrics 161
7.2.3. Depth Computation and Visualization 163
7.3. Results 163
7.3.1. Robust and Uncertainty-Aware Consensus Extraction Using Inclusion Depth 163
7.3.2. Multi-Modal Analysis and Interactive Candidate Selection 164
7.4. Discussion 168

8. Conclusion 173

A. Implementation of Delineation Error Detection Systems in Time-Critical Radiotherapy: Do AI-Supported Optimization and Human Preferences Meet? 181

B. Inclusion Depth for Contour Ensembles 183

C. LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles 187

Acknowledgements 195

Curriculum Vitæ 199

List of Publications 201

SUMMARY

Radiotherapy (RT) is a widespread and effective technique to treat cancers by killing cancerous cells with rays of radiation. Building upon advances in image guidance and dose delivery technology like Proton Therapy, Adaptive RT promises more effective tumor decimation and a reduction of the incidence and severity of side effects. Unfortunately, the clinical implementation of adaptive workflows is challenging due to their resource-intensive nature. Therefore, their successful adoption lingers on overcoming several bottlenecks in the treatment planning process.

In this dissertation, we focus on methods used for the image segmentation or contouring step, which allows the localization of the anatomical structures required for dose optimization and evaluation. Until recently, clinicians had to manually delineate dozens of organs-at-risk and target volumes across hundreds of slices of the patient's three-dimensional images. A process that is extremely time-consuming. The advent of deep learning-based artificial intelligence (AI) has changed the landscape: a modern auto-segmentation AI can produce segmentations for most of a patient's anatomy in minutes.

Despite increasing automation in the segmentation process, it remains time and resource-intensive. Due to the segmentations' criticality for the patient's outcome and the errors the AI will commit, clinicians must perform a quality assessment of the AI's outputs. Depending on the case's complexity, the duration of the quality assessment process can negate the time gains auto-segmentation tools bring.

Deep ensemble AIs represent an advancement in medical image segmentation. Instead of providing a deterministic output, deep ensemble AIs produce a set of plausible candidates that aim to model inter-clinician annotation variability. Consensus segmentations obtained from ensembles tend to be more accurate and robust than the single-prediction deterministic counterpart. Nevertheless, by only using the consensus, a lot of potentially useful information is being discarded.

In this dissertation, we contribute to different phases of the segmentation quality assessment process. We characterize this process and introduce methods that leverage the raw outputs of deep ensemble AIs to support and speed up quality assessment tasks. The methods presented show new ways of analyzing and using ensembles in RT. Nevertheless, since these are relevant outside RT, we keep the presentation of the methods general and evaluate them in other application scenarios, such as the analysis of simulation ensembles or meteorological data.

Before fixing segmentation failures, clinicians must find them. This process can be time-consuming and fatiguing when failures are sparse and spread through the patient's three-dimensional images. We present and evaluate a delineation error detection system, which guides clinicians to slices of three-dimensional images that contain potentially clinically relevant segmentation failures. We co-designed the DEDS with clinicians and refined it based on an observational study, which allowed us to characterize clinicians' navigation patterns and the use of information sources like AI uncertainty and patients' dose distributions. We evaluated the DEDS' potential to speed up the QA process through a simulation study with a retrospective cohort of patients. Results indicate that speed-ups are the most significant when equipping the DEDS with information sources indicative of clinical priority, which prevents unnecessary edits.

Visual inspection of the segmentation ensemble permits understanding the main trends and detecting anomalies that might indicate segmentation failures. Using a spaghetti plot to visualize all ensemble members is straightforward but prone to clutter. Contour boxplots prevent clutter and extra complexity by distilling essential ensemble information, which permits more efficient ensemble inspection. Nevertheless, they are time-consuming to compute, reducing their practical value. We present Inclusion Depth for contour ensembles. Inclusion Depth yields per ensemble member centrality scores that allow characterizing the distribution of segmentation ensembles in terms of properties like the median, trimmed mean, confidence bands, and outliers. Compared to previous contour depth notions, Inclusion Depth is significantly faster, making it more applicable in practice for time-critical contexts like QA in adaptive RT. We show how Inclusion Depth permits creating contour boxplots for ensembles with hundreds of segmentations in seconds.

It is not uncommon for distinct representative shapes to co-occur within a contour ensemble. With ensembles created by clinicians, for instance, different institutions, training sessions, or experience levels can lead to distinct shapes (i.e., modes of variation) for the same structure. When trained on these data, deep ensemble AIs would yield similarly multi-modal ensembles. In quality assessment, being able to extract these representatives would pave the way for new ensemble-based interactive segmentation workflows. Applying traditional contour depth notions to these multi-modal ensembles collapses the existing variation modes and can lead to uninformative centrality scores. To address this issue, we present the first framework for multi-modal contour depth, which also includes notable runtime improvements for depth computation. When used with Inclusion Depth, multi-modal contour depth permits clustering the different modes of variation and determining cluster-dependent scores that appropriately characterize the data. Variation modes can be then independently analyzed using uni-modal depth machinery like contour boxplots.

The global perspective of contour depth methods, which consider the entire volume, may be insufficient when parts of the contours are noisy or when the resolution of the ensemble is too large to process within a reasonable time. Correlation clustering methods provide a solution by partitioning the spatial domain of the ensemble into highly correlated regions that can be used to localize analyses. Existing correlation clustering algorithms do not scale well as the resolution of the ensemble increases. We introduce the Local-to-Global Correlation Clustering (LoGCC) method, which partitions the ensemble's spatial domain into coarser primitives, representing areas of consistent ensemble member behavior. Unlike previous correlation clustering methods, the proposed LoGCC achieves significantly faster runtimes by leveraging the ensemble's spatial structure and decoupling computations into local and global steps. Like with Inclusion Depth, these speed gains enable LoGCC to analyze large datasets in time-critical fields such as adaptive radiotherapy (RT).

Throughout this dissertation, our approach focused on designing modular, flexible analysis methods applicable across different tasks and domains. We demonstrate how the delineation error detection system, multi-modal Inclusion Depth, and Local-to-Global Correlation Clustering support quality assessment in RT and extend to fields like meteorology. We also speculate on their potential as foundational elements for more complex workflows. For example, extracted modes of variation, which indicate representative shapes in the ensemble, could be repurposed as an interactive segmentation tool. Alternatively, consistent regions detected by correlation clustering could be used as building blocks to enable localized contour analysis and editing.

We hope the proposed contour ensemble visual analysis methods inspire the development of more efficient analysis workflows that harness ensembles' power in RT and beyond.

SAMENVATTING

Radiotherapie (RT) is een wijdverbreide en effectieve techniek om kanker te behandelen door kankercellen te vernietigen met stralingsbundels. Dankzij vooruitgangen in beeldgeleiding en dosisafgifte-technologieën, zoals protonentherapie, biedt adaptieve RT de mogelijkheid om tumoren effectiever te vernietigen en de incidentie en ernst van bijwerkingen te verminderen. Helaas is de klinische implementatie van adaptieve werkprocessen uitdagend vanwege hun arbeidsintensieve aard. Het succesvol toepassen ervan vereist het overwinnen van verschillende knelpunten in het behandelplanningsproces.

In dit proefschrift richten we ons op methoden die worden gebruikt voor de beeldsegmentatie- of contoureringsstap, waarmee anatomische structuren worden gelokaliseerd die nodig zijn voor dosisoptimalisatie en -evaluatie. Tot voor kort moesten clinici tientallen risico-organen en doelvolumes handmatig afbakenen over honderden plakjes van de driedimensionale beelden van de patiënt – een proces dat enorm tijdrovend is. De opkomst van op deep learning gebaseerde kunstmatige intelligentie (AI) heeft dit landschap veranderd: een moderne auto-segmentatie-AI kan binnen enkele minuten de meeste anatomische structuren van een patiënt segmenteren.

Ondanks de toegenomen automatisering van het segmentatieproces blijft het tijd- en arbeidsintensief. Vanwege het cruciale belang van de segmentaties voor het behandelresultaat van de patiënt en de fouten die de AI kan maken, moeten clinici een kwaliteitscontrole uitvoeren op de outputs van de AI. Afhankelijk van de complexiteit van het geval kan de duur van dit kwaliteitscontroleproces de tijdwinst die autosegmentatietools opleveren, tenietdoen.

Deep ensemble AIs vormen een vooruitgang in medische beeldsegmentatie. In plaats van een deterministische output te leveren, genereren deep ensemble AIs een set plausibele kandidaten die de variabiliteit in annotaties tussen clinici modelleren. Consensussegmentaties verkregen uit ensembles zijn doorgaans nauwkeuriger en robuuster dan de deterministische tegenhangers. Toch wordt door uitsluitend de consensus te gebruiken veel potentieel nuttige informatie genegeerd.

In dit proefschrift dragen we bij aan verschillende fasen van het kwaliteitscontroleproces van segmentaties. We karakteriseren dit proces en introduceren methoden die de ruwe outputs van deep ensemble AIs benutten om kwaliteitscontrole taken te ondersteunen en te versnellen. De voorgestelde methoden bieden nieuwe manieren om ensembles te ana-

lyseren en te gebruiken in RT. Omdat deze methoden ook buiten RT relevant zijn, presenteren we ze in algemene termen en evalueren we ze in andere toepassingsscenario's, zoals de analyse van simulatie-ensembles of meteorologische gegevens.

Voor clinici segmentatiefouten kunnen herstellen, moeten ze deze eerst vinden. Dit proces kan tijdrovend en vermoeiend zijn wanneer fouten schaars zijn en verspreid liggen door de driedimensionale beelden van de patiënt. We presenteren en evalueren een delineation error detection system (DEDS) dat clinici begeleidt naar plakjes van driedimensionale beelden die mogelijk klinisch relevante segmentatiefouten bevatten. Het DEDS is samen met clinici ontworpen en verfijnd op basis van een observatiestudie, waarmee we het navigatiegedrag van clinici en hun gebruik van informatiebronnen zoals AI-onzekerheid en dosisdistributies van patiënten karakteriseerden. Uit een simulatiestudie met een retrospectieve patiënten cohort blijkt dat de grootste tijdswinst wordt behaald wanneer het DEDS wordt uitgerust met informatiebronnen die klinische prioriteit aangeven, waardoor onnodige bewerkingen worden voorkomen.

De visuele inspectie van een segmentatie-ensemble biedt inzicht in de belangrijkste trends en maakt het mogelijk om anomalieën te detecteren die op segmentatiefouten kunnen wijzen. Een spaghetti-plot van alle ensembleleden is een eenvoudige visualisatie, maar kan snel chaotisch worden. Contourboxplots voorkomen rommel en extra complexiteit door essentiële ensemble-informatie te distilleren, waardoor een efficiëntere inspectie mogelijk is. Het berekenen van contourboxplots is echter tijdrovend, wat hun praktische waarde vermindert. We introduceren Inclusion Depth voor contour-ensembles. Inclusion Depth levert centrale scores per ensemblelid waarmee de verdeling van segmentatie-ensembles kan worden gekarakteriseerd op basis van eigenschappen zoals de mediaan, bijgesneden gemiddelden, betrouwbaarheidsbanden en uitschieters. Vergeleken met eerdere contour depth-concepten is Inclusion Depth aanzienlijk sneller, wat het toepasbaar maakt in tijdkritieke contexten zoals kwaliteitscontrole in adaptieve RT. We tonen aan hoe Inclusion Depth contourboxplots kan genereren voor ensembles met honderden segmentaties binnen enkele seconden.

Het is niet ongebruikelijk dat verschillende representatieve vormen voorkomen binnen een contour-ensemble. Bij ensembles gemaakt door clinici kunnen bijvoorbeeld verschillende instellingen, trainingssessies of ervaringsniveaus leiden tot variatie in de vormen (d.w.z. variatiemodi) voor dezelfde structuur. Wanneer getraind op deze gegevens, zouden deep ensemble AIs vergelijkbare multi-modale ensembles opleveren. In kwaliteitscontrole kan het extraheren van deze representatieve vormen de weg vrijmaken voor nieuwe ensemble-gebaseerde interactieve segmentatieworkflows. Traditionele contour depth-concepten toegepast op deze multi-modale ensembles negeren bestaande variatiemodi, wat kan leiden tot niet-informatieve centrale scores. Om dit probleem aan te pakken, presenteren we het eerste raamwerk voor multi-modale contour depth, dat ook aanzienlijke verbeteringen biedt in de rekentijd. Wanneer gebruikt met Inclusion Depth, maakt multi-modale contour depth het mogelijk om de verschillende variatiemodi te clusteren en cluster-afhankelijke scores te bepalen die de gegevens passend karakteriseren. Variatiemodi kunnen vervolgens onafhankelijk worden geanalyseerd met behulp van uni-modale depth-methoden zoals contourboxplots.

Het globale perspectief van contour depth-methoden, waarbij het volledige volume wordt beschouwd, kan ontoereikend zijn bij ruisende contouren of te grote ensemble-resoluties. Correlatie-clusteringmethoden bieden een oplossing door het ensemble te partitioneren in sterk gecorreleerde regio's. Bestaande correlatie-clusteringalgoritmen schalen echter slecht bij toenemende ensemble-resolutie. We introduceren de Local-to-Global Correlation Clustering (LoGCC)-methode, die de ruimtelijke structuur van het ensemble benut om snellere looptijden te realiseren door berekeningen lokaal en globaal te scheiden. Zoals bij Inclusion Depth maken deze snelheidsverbeteringen LoGCC toepasbaar in tijdkritieke contexten zoals adaptieve RT.

In dit proefschrift was onze aanpak gericht op het ontwerpen van modulaire, flexibele analysemethoden die toepasbaar zijn in verschillende taken en domeinen. We laten zien hoe het DEDS, multi-modal Inclusion Depth, en LoGCC de kwaliteitsbeoordeling in RT ondersteunen en uitbreiden naar gebieden als meteorologie. We speculeren ook over hun potentieel als basiselementen voor complexere workflows. Zo zouden bijvoorbeeld geëxtraheerde variatiemodi, die representatieve vormen in het ensemble aangeven, opnieuw kunnen worden gebruikt als interactief segmentatiehulpmiddel. Als alternatief zouden consistente regio's, gedetecteerd door correlatieclustering, gebruikt kunnen worden als bouwstenen voor gelokaliseerde contouranalyse en -bewerking.

We hopen dat de voorgestelde visualisatiemethoden voor contourensembles inspireren tot de ontwikkeling van efficiëntere analysemethoden die de kracht van ensembles benutten in RT en daarbuiten.

1

INTRODUCTION

In 1895, Wilhelm Conrad Röntgen discovered X-rays [1]. Shortly after, radiation therapy (RT) was born. In RT, cancer patients get a dose of radiation directed at the tumor with the intent of killing the malignant cells within. Over the course of the twentieth century, RT consolidated as one of humanity's main tools against cancer [2].

Over the last fifty years, numerous technological and practical innovations have significantly enhanced the precision and safety of radiotherapy (RT). Fractionating the radiation dose allows for effective and sustained tumor control while enabling surrounding organs-at-risk (OARs) to recover between treatments [3]. The integration of imaging modalities, such as computed tomography and magnetic resonance imaging, during treatment has driven the development of adaptive RT. Adaptive RT leverages image guidance to monitor patient anatomy in real time and adapt treatment plans to account for changes such as tumor shrinkage or anatomical shifts [4]. Finally, advanced dose delivery techniques, including proton-based therapy, exploit the unique physical properties of protons to create sharp dose gradients that precisely target the tumor while sparing adjacent healthy tissues [5].

Adaptive therapies, combined with highly precise dose delivery mechanisms such as proton therapy, are poised to become the standard of care in radiotherapy. However, their practical adoption faces challenges in reducing their resource demands [6–8].This dissertation contributes to streamlining a crucial step in the RT workflow known as segmentation. This step produces three-dimensional contours of tumors and OARs that allow their identification and localization. Before elaborating on the performance challenges the segmentation step poses and our contributions, we detail the clinical context of the adaptive RT workflow.

1.1. CLINICAL CONTEXT: ADAPTIVE RADIOTHERAPY

The radiation dose is not delivered all at once. Instead, the RT treatment is fractionated over days or weeks, depending on the type of therapy. During this period, the patient's anatomy can change significantly; for example, tumors may shrink, and organs may enlarge or shift position. Traditional RT compensates for such geometric changes and other uncertainties in dose delivery by employing safety margins around the tumor [9]. However, these margins often come at the expense of impacting the ability of OARs to heal and regenerate after irradiation. Adaptive RT addresses this limitation by monitoring the patient's anatomy and updating the treatment plan as anatomical changes occur, allowing for tighter margins and reduced side effects [4].

Adaptive RT implementations vary across clinics, with adaptation strategies categorized based on frequency: offline (days/hours), online (minutes), and real-time (seconds and milliseconds). The choice of strategy ideally depends on the frequency of anatomical changes [6] but is often constrained by the significant resource demands associated with more responsive schemes, which require performing complex processes like image acquisition, segmentation, and dose calculations in reduced time frames [8]. Consequently, there is a general need to streamline adaptive workflows by reducing these processes' resource footprint to make them feasible in clinical practice.

To illustrate the clinical adaptive RT workflow, we use Holland Proton Therapy Center (HollandPTC) in the Netherlands as an example. Patients at centers like HollandPTC benefit most from improved workflow efficiency and higher adaptation frequencies, as proton beams' precision synergizes with tight safety margins. HollandPTC implements an offline strategy to adapt the patients' treatment plans when their anatomy changes. Based on the literature and discussions with medical professionals at Leiden University Medical Center and Utrecht Medical Center, we observed that HollandPTC's workflow is representative of other centers implementing adaptive RT [10–12].

Figure 1.1 presents a simplified depiction of HollandPTC's RT workflow. The workflow starts with a treatment planning session to gather the necessary materials and information for treating the patient. After the treatment plan has been approved, a sequence of sessions ensues, where the patient receives dose fractions spread over several weeks in a gantry, a robotized arm paired with a particle accelerator capable of shooting carefully calibrated radiation beams toward the tumor. Inset (b) in the figure shows the gantry at HollandPTC. After a given amount of time has elapsed or certain anatomical changes have been observed, the offline adaptation/treatment planning will be triggered and performed parallel to the fractions. Once the new treatment plan is available, it will replace the previous one.

Inset (a) of Figure 1.1 details the core steps of the treatment planning process. Starting with the simulation step, clinicians acquire images of the patient's anatomy and gather additional necessary information to construct fixation devices and simulate the distribution of the radiation dose. In the subsequent registration and segmentation steps, clinicians combine the images and use them to delineate tumors and OARs in three-dimensional space. Finally, a treatment planning system steered by clinicians produces a gantry configuration to deliver a clinically suitable dose based on the segmentations and predefined dosimetric optimization goals. Clinicians follow the ALARA (i.e., as low as reasonably achievable) principle to balance tumor coverage and OAR sparing when defining optimization objectives and constraints [13]. Inset (b) of Figure 1.1 illustrates HollandPTC's gantry used to administer the dose fraction to the patients.

Offline adaptations entail similar steps to the treatment planning process. Nevertheless, these can run faster by leveraging prior patient information like segmentations, which can be automatically transferred to the new images and adjusted instead of generating them from scratch, and the dosimetric configuration. Higher frequency adaptations behave similarly but must run in a significantly shorter time frame [14].

Treatment planning is time and resource-intensive, which limits the availability of current RT treatments and the practical implementation of adaptive RT. The first treatment planning session can take clinicians from different specialties hours or days to complete [12]. Subsequent adaptation sessions profit from reusing prior patient information. Nevertheless, they can still take hours, which is likely too long. On the one hand, the patient's anatomy might have changed by the end of the re-planning, invalidating the adaptation. On the other hand, performing this hour-long still-clinician-intensive process creates an immense burden on clinical infrastructure, reducing its capacity to treat patients [6]. Therefore, there is a need to increase treatment planning efficiency.

Artificial Intelligence (AI) offers a promising solution by automating many steps in RT, potentially improving efficiency and scalability. Yet, when AI systems fail—whether due to errors or uncertainties—clinicians must intervene, negating the efficiencies gained. As we will see next, the segmentation step is a clear example of this duality. Auto-segmentation technologies have dramatically accelerated the process. However, painstakingly inspecting and correcting errors in the generated segmentations is time-consuming and fatiguing for clinicians. To fully realize AI's potential, it is essential to foster effective human-AI collaboration, ensuring reliability and trust in clinical practice. This dissertation contributes to this vision by addressing the visualization and interaction challenges that arise when clinicians must inspect the outputs of auto-segmentation AIs.

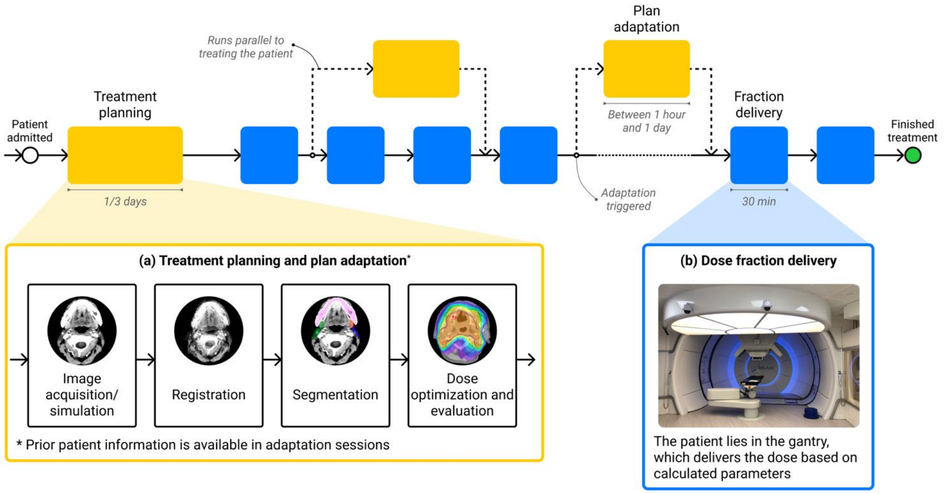


Figure 1.1.: Simplified depiction of offline-adaptive PT workflow at HollandPTC in the Netherlands. The dose delivery starts after completing the treatment planning process (yellow). The patient receives the dose in fractions (blue) split spread over several weeks. If changes are observed or a given amount of time has passed, an offline plan adaptation is triggered. The treatment planning and plan adaptation processes consist of image acquisition/simulation, registration, segmentation, and dose optimization and evaluation subprocesses.

1.2. MEDICAL IMAGE SEGMENTATION

An effective approach to streamline and accelerate treatment planning and plan adaptation is to improve individual steps of the workflow. We focus on the segmentation step, which identifies and localizes the patient's tumors and OARs. The resulting contours are then used for dose optimization and evaluation. Despite significant advances in automation technologies, segmentation can still take up to an hour of clinicians' time, as they must check and correct the generated contours. Therefore, segmentation remains a time-consuming process that can lead to clinician fatigue.

Figure 1.2 presents an overview of the segmentation step. It receives as input the patient's available three-dimensional images and a list of structures to delineate. The primary image is usually a computerized tomography (CT) scan, later used for dose calculation. Clinicians use available secondary images like positron emission technology (PET)-CT and magnetic resonance imaging (MRI) scans to resolve ambiguity in areas where the CT lacks contrast, like soft tissue. The outputs of the segmentation step are the delineations of the structures, represented as three-dimensional images whose voxels (i.e., three-dimensional pixels) have a label that indicates the presence of an anatomical structure of interest at a given location.

Manually segmenting tumors and OARs is time-consuming and fatiguing. In practice, three-dimensional images are represented as stacks of two-dimensional slices. Therefore, clinicians must inspect potentially hundreds of slices and delineate segments on many of them [15]. A contour might already exist (e.g., as a result of a semi-automatic tool like between slice interpolation or existing previously generated delineations). In this case, clinicians continue with the segmentation quality assessment (QA) process, which can involve manual tweaking if an error is present. The process repeats until all slices from all structures of interest have been segmented.

The number of anatomical structures to be segmented, their variation of sizes, and the complexity of their shapes further increase the segmentation workload. For instance, treatments in the head-and-neck area can involve delineating dozens of structures of varying characteristics [16]. The mandible spans many slices but is easy to segment using automatic tools because its bony material surrounded by soft tissue produces a crisp boundary in the CT image. In contrast, structures like the optic system and the swallowing muscles often require additional images or expert knowledge due to their poor contrast in CT images [17].

1.3. AI-SUPPORTED SEGMENTATION

Efficient segmentation workflows, enhanced training of clinicians and delineation standards, and leveraging automation can increase efficiency and reduce the workload [18]. This dissertation focuses on automated medical image segmentation, which has recently been disrupted, like many other fields, by AI.

A transformative milestone in the history of medical image segmentation was the rise of deep learning, a subset of AI that excels at learning from data [19]. Before the advent of deep learning-based auto-segmentation AIs, registration-based techniques like model and atlas-based auto-segmentation [15] constituted the state-of-the-art. Although powerful, their effectiveness depends on carefully configured parameters and the availability of high-quality or patient-specific datasets [20].

Deep learning-based approaches leverage modular building blocks like convolutions and retain task-specific knowledge in the form of parameters automatically learned from data [21, 22]. During training, the AI is exposed to labeled examples from a dataset. In the case of segmentation, these correspond to image/segmentation pairs. Once

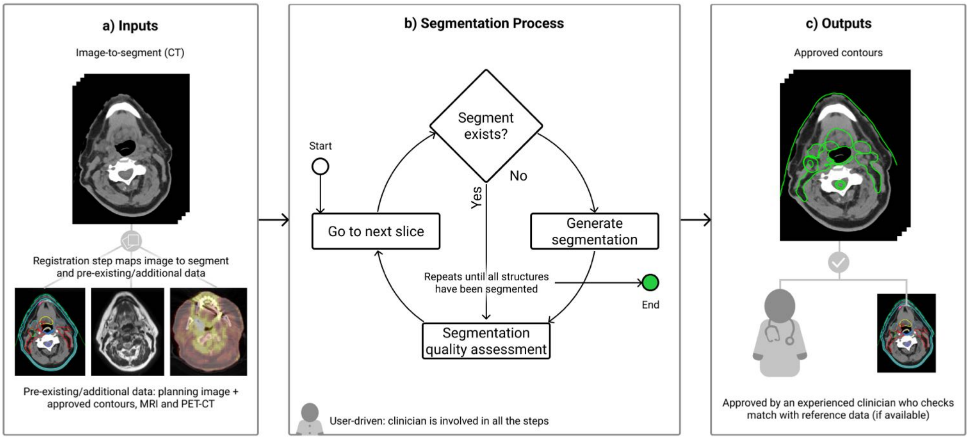

a) Inputs
Image-to-segment (CT)
Registration step maps image to segment and pre-existing/additional data
Pre-existing/additional data: planning image + approved contours, MRI and PET-CT
b) Segmentation Process
Start
Segment exists?
Yes
No
Go to next slice
Generate segmentation
Repeats until all structures have been segmented
End
Segmentation quality assessment
User-driven: clinician is involved in all the steps
c) Outputs
Approved contours
Approved by an experienced clinician who checks match with reference data (if available)

Figure 1.2.: Medical image segmentation process. The inputs (a) are several three-dimensional images: a primary image, usually a CT scan, and potentially other secondary images, like PET-CT or MRI scans, that can help resolve ambiguity. The segmentation process (b) consists of a loop of steps that must be repeated until all structures have been segmented. The outputs (c) are the segmentations of structures of interest.

trained, the model can be deployed on new, unseen data. Ongoing research and data-gathering efforts keep improving segmentation models' ability to generalize beyond the training set.

The introduction of AI has transformed the segmentation workflow. As Figure 1.3 illustrates, the AI-supported workflow consists of AI segmentation generation and clinician-driven segmentation QA stages. These steps closely match those of traditional segmentation (Figure 1.2), but have been reorganized so that the AI performs the bulk of the manual contour drawing work. In the QA stage, clinicians navigate through the three-dimensional auto-segmented images, looking for and editing delineation errors.

Modern auto-segmentation AIs bring impressive performance gains, segmenting tens of structures in minutes. Unfortunately, full automation remains elusive because AI methods struggle to obtain clinically acceptable results, requiring clinician input and oversight [20]. The QA stage is demanding for clinicians and, although faster than manual segmentation from scratch, it is not fast enough for adaptive RT workflows. Therefore, there is a need for tools that facilitate the inter-operation of clinicians and AI, leading to more effective and efficient QA.

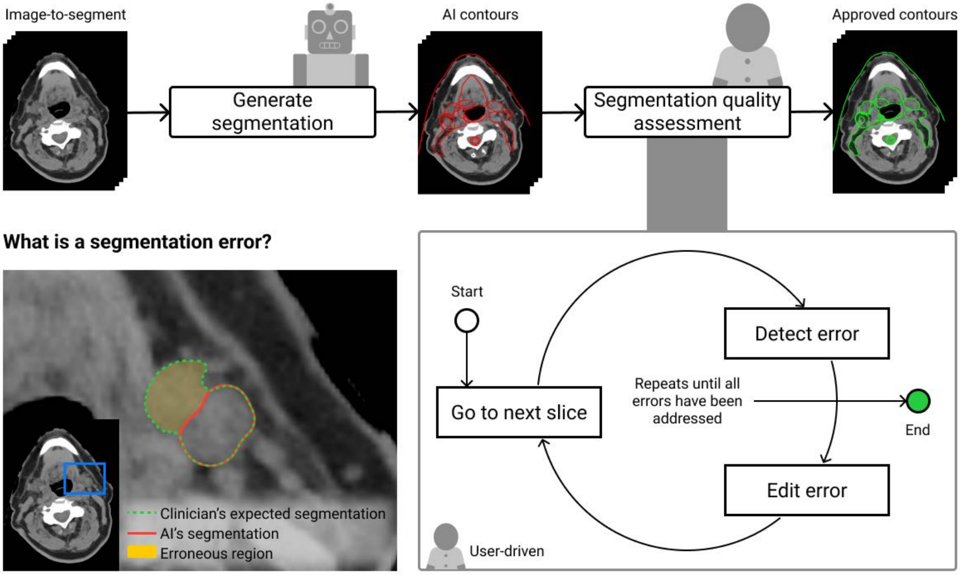


Figure 1.3.: AI-supported segmentation process. Inputs and outputs remain the same with respect to the traditional segmentation process in Figure 1.2. Steps in the process are re-distributed among clinicians and AI.

1.4. DEEP ENSEMBLE AI

A recent development relevant to this dissertation is that of deepensemble AIs [23–26]. Traditional deterministic AIs predict a single contour per structure. In contrast, deep ensemble AIs yield an ensemble or set of plausible segmentations.

Deep ensemble AIs are motivated by clinicians' behavior. If, as Figure 1.4 exemplifies, one asks different clinicians to segment the same structure, their delineations (green and blue lines) are likely to differ in regions where image information is not enough or delineation instructions are not clear. A similar outcome can be observed when asking the same clinician to delineate a structure at different time points. Deep ensembles aim to model these inter/intra-observer variabilities [27, 28] by producing an ensemble that reflects clinicians' differing viewpoints.

Having an ensemble of opinions from clinicians is valuable for several reasons. First, ensembles can lead to more accurate and robust segmentations due to the power of consensus [29]. Second, ensembles permit quantifying inter/intra-observer variabilities (depicted as a yellow shaded area in Figure 1.4 for a portion of the image). These variabilities

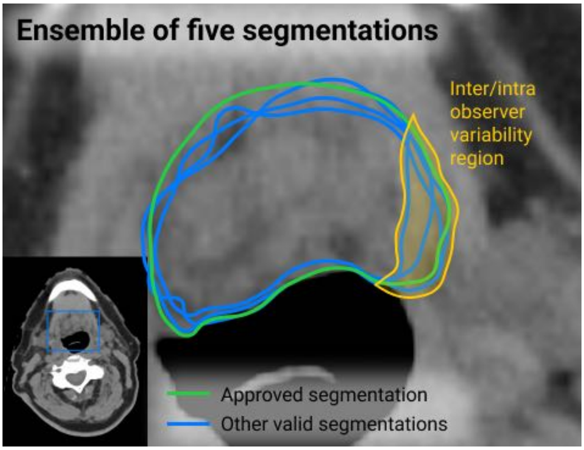

Ensemble of five segmentations
Inter/intra observer variability region
Approved segmentation
Other valid segmentations

Figure 1.4.: Example of inter/intra-observer variability (IOV) for an axial slice of a head-and-neck CT image. Five annotators produced segmentations for the palate. The green line corresponds to the approved segmentation and the blue lines exemplify different medical opinions. The shaded yellow region exemplifies the IOV region for one section of the contours.

permit estimating segmentations' geometric uncertainty, which in turn can be used to detect the areas where segmentation errors are most likely to occur and to inform the definition of safety margins used during treatment to ensure tumor coverage and OARs protection [9, 30, 31].

Similarly to clinicians' ensembles, deep ensembles yield more robust predictions and allow quantifying uncertainty [32]. Crucially, deep ensemble AIs operate at a fraction of the time it would take to consult multiple clinicians, potentially unlocking novel clinician-AI interactions in the quality assessment process. For instance, the uncertainty derived from the ensemble could guide clinicians towards part of the segmentations requiring attention [33, 34]. Ensembles could also support the editing process by allowing clinicians to select the desirable segmentation among the pool of candidates, potentially a quicker alternative than most currently available scribble-based interactive segmentation refinement methods.

We conclude this section with a note of caution regarding the use of deep ensembles. Consensuses of deep ensembles perform well in practical segmentation tasks. Nevertheless, other ensemble features like individual members and their variability can be more challenging to understand and harness. This difficulty arises from deep ensemble AIs' tendency to entangle data and model uncertainties, produce implausible segmentations, and exhibit reduced segmentation diversity [23, 35]. Addressing these challenges has become a focus of significant research efforts [25, 26]. We anticipate the techniques presented in this dissertation will benefit from ongoing advancements in deep ensemble AI technology.

1.5. THESIS OUTLINE

The clinical adoption of deep ensembles beyond consensus-based auto-segmentation remains limited due to a lack of understanding of their potential to support the quality assessment process. Successfully integrating deep ensembles into other aspects of AI-supported segmentation requires a thorough understanding of existing workflows and the challenges where ensemble information can be most beneficial.

In the first part of this dissertation (Chapters 2 and 3), we present user studies that clarify the role of deep ensembles in clinical workflows. By studying clinicians from several cancer treatment centers in the Netherlands, we identified key aspects of the quality assessment process, particularly during error detection and correction. These studies underscored the importance of ensemble analysis, a process that clinicians rely on to effectively use and adopt deep ensemble AI systems.

Ensemble analysis is central to quality assessment, enabling clinicians to leverage ensemble information to identify errors and interpret geometric uncertainties. For instance, understanding representative contours and outliers within the ensemble allows clinicians to differentiate between segmentation errors and expected variability. Additionally, recognizing distinct representative shapes in the ensemble supports selection-based editing during the correction process.

However, visually analyzing deep ensembles is challenging due to their multi-dimensional nature, which includes spatial dimensions and the ensemble dimension, often resulting in datasets with hundreds of ensemble members and millions of voxels. Existing methods, such as contour boxplots [36] and correlation clustering [37], help manage this complexity but struggle with large datasets and often provide only global insights, lacking localized perspectives.

In the second part of this dissertation (Chapters 4–6), we propose efficient methods for visually analyzing deep ensembles, motivated by the time-critical nature of adaptive radiotherapy. These techniques simplify ensemble data to facilitate task-specific inspection, making them more accessible to clinicians. While the methods are tailored to radiotherapy, they are broadly applicable to other domains involving contour or scalar field ensembles, such as meteorological forecasting. For example, we demonstrate how these techniques can distill and visualize relevant information from meteorological forecast ensembles to support decision-making.

To summarize, this dissertation contributes to streamlining the AI-supported segmentation process in radiotherapy by exploring how deep ensembles can enhance quality assessment and by introducing efficient methods for ensemble analysis. The following subsections provide a chapter-wise overview of our contributions. Given the diversity of research questions and methodologies addressed, each chapter is self-contained and includes its own background and related work.

1.5.1. TOWARDS FAST HUMAN-CENTRED CONTOURING WORKFLOWS FOR ADAPTIVE EXTERNAL BEAM RADIOTHERAPY

In **Chapter 2**, we investigate the medical image segmentation workflow in the context of adaptive RT. We conducted an observational study in several Dutch RT clinics to characterize clinical segmentation workflow, potential bottlenecks, and speed-up opportunities. The study uncovered the slice navigation and segmentation error analysis quality assessment components explained earlier in the introduction and revealed three context-dependent variables that affect segmentation performance: usable additional information, applicable domain-specific knowledge, and available capabilities in segmentation software. Clinicians leverage these variables to reduce segmentation time and effort by determining which areas need attention and modulating the precision level required while editing.

This chapter is based on the published paper:

> Chaves-de-Plaza, N. F., Mody, P., Hildebrandt, K., Staring, M., Astreinidou, E., de Ridder, M., de Ridder, H. & van Egmond, R. "Towards fast human-centred contouring workflows for adaptive external beam radiotherapy". In: Proceedings of the Human Factors and Ergonomics Society Europe. 2022, pp. 111-31

1.5.2. IMPLEMENTATION OF DELINEATION ERROR DETECTION SYSTEMS IN TIME-CRITICAL RADIOTHERAPY: DO AI-SUPPORTED OPTIMIZATION AND HUMAN PREFERENCES MEET?

Chapter 3 delves into the navigation and error detection QA steps, focusing on the clinical adoption of delineation error detection systems (DEDS). DEDS can spare clinicians from unnecessary navigation and editing workloads by directing their attention to slices that contain segmentation anomalies. Typically, DEDS only consider error metrics and are developed separately from the auto-segmentation technologies, incurring elevated development costs. This chapter presents a DEDS that can help clinicians find clinically significant segmentation anomalies in deep ensemble AI-generated segmentations using only the ensemble and clinical information sources like the dose. We leveraged user studies to inform the system's development, and to learn about users' workflows and information needs. This chapter also tackles the critical lingering question of DEDS' potential to speed up segmentation QA in clinical practice. To this end, we compared several DEDS workflows using a simulation study on a retrospective cohort of head and neck cancer patients. Results indicate that DEDS' time-reducing potential depends on the availability of clinical prioritization metrics, the possibility of trading off segmentation accuracy without affecting the patient's treatment, and the implementation of effective user interfaces to guide users through DEDS' results.

By analyzing clinician workflows, Chapters 2 and 3 provide foundational knowledge of the quality assessment workflow like the presence of the underlying ensemble analysis process. We use the uncovered challenges for visually analyzing deep ensembles to motivate the techniques presented in subsequent chapters.

This chapter is based on the published paper:

> Chaves-de-Plaza, Nicolas F., P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, A. Vilanova, and R. van Egmond. "Implementation of delineation error detection systems in time-critical radiotherapy: Do AI-supported optimization and human preferences meet?" In: Cognition, Technology & Work (2024). doi: 10.1007/s10111-024-00784-4

1.5.3. INCLUSION DEPTH FOR CONTOUR ENSEMBLES

Contour Boxplots can help to reduce the user's analysis burden but are time-consuming to compute. Contour boxplots offer a simplified depiction of the segmentation ensemble, highlighting statistical features like its representatives (e.g., the median or mean contours), confidence intervals, and outliers [36]. Contour boxplots need for their construction a centrality score for each segmentation's contour known as contour depth, which indicates how much of an inlier or an outlier that segmentation is. Computing contour depths is a time-consuming process that scales poorly as the number of ensemble members grows, limiting the adoption of contour boxplots and other techniques that rely on depth scores. **Chapter 4** presents Inclusion Depth, a new contour depth notion that processes large segmentation ensembles in seconds. Inclusion Depth offers several theoretical guarantees and is simple to implement and understand, relying on a simple principle of using inside/outside relationships between contours to assess their centrality. Finally, Inclusion Depth is significantly faster than previous contour depth notions like Contour Band Depth while yielding depth estimates of comparable quality.

This chapter is based on the published paper:

> Chaves-de-Plaza, Nicolas F., P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "Inclusion Depth for Contour Ensembles". In: IEEE Transactions on Visualization and Computer Graphics 30.9 (2024), pp. 6560–6571. doi: 10.1109/TVCG.2024.3350076

1.5.4. DEPTH FOR MULTI-MODAL CONTOUR ENSEMBLES

The following chapter tackles a limiting assumption of existing contour depth methodologies that limits their practical utility. Namely, they assume that all contours in an ensemble are drawn from the same distribution or exhibit one mode of variation. In reality, the contour ensembles might consist of multi-modal distributions arising due to discrepancies in the data gathering and modeling processes. For instance, clinics and clinicians can disagree on a structure's delineation, especially if images contain ambiguities. Deep ensemble AIs trained with these data would yield predictions exhibiting similar multi-modal patterns. As a practical example, imagine two modes of variation arising for the case in Figure 1.4: one portion of the segmentations could underestimate the palate's area, and the remaining could overestimate it. Uni-modal contour depth methodologies and derived analyses like contour boxplots would misrepresent these variation modes by collapsing them into one with misleading statistical features.

To address this issue, **Chapter 5** presents a multi-modal version of the contour depth methodology that permits analyzing ensembles with several modes of variation. First, we introduce the concept of relative contour depth to quantify the depth of a contour with respect to a subset of the ensemble. Second, we propose a clustering algorithm, CDclust, that leverages relative depth to disentangle modes of variation. Given that the algorithm tries different configurations of the expected variation modes, it requires many depth evaluations. To permit using CDclust interactively, we further streamline Inclusion Depth computation by introducing a linear time algorithm and the inclusion matrix, which allows fast computation and re-computation of contour depth notions. CDclust partitions the ensemble into separate variation modes, which clinicians can inspect using a multi-modal contour boxplots. To further increase multi-modal Inclusion Depth's practical utility, we developed and open-sourced a Python library that permits using multi-modal contour depth techniques in any contour dataset.

This chapter is based on the published paper:

> Chaves-de-Plaza, Nicolas F., M. Molenaar, P. Mody, M. Staring,

R. van Egmond, E. Eisemann, A. Vilanova, and K. Hildebrandt. "Depth for Multi-Modal Contour Ensembles". In: Computer Graphics Forum 43.3 (2024), e15083. doi: 10.1111/CGF.15083

1.5.5. LOGCC: LOCAL-TO-GLOBAL CORRELATION CLUSTERING FOR SCALAR FIELD ENSEMBLES

Visual ensemble analysis methods, like Inclusion Depth and its multi-modal extension, typically operate across the entire spatial grid. However, this approach can introduce challenges. Analyzing the full spatial domain may include irrelevant areas, introducing noise or artifacts, while large grids can make computations time-consuming and impractical. This chapter explores partitioning the ensemble's spatial domain to localize and potentially streamline ensemble analyses.

There are several approaches to partitioning an ensemble's spatial domain into consistent regions. **Chapter 6** focuses on Correlation Clustering, favored for its conceptual simplicity, reduced memory footprint, and ability to uncover connections between disjoint regions. However, existing correlation clustering methods are often time-consuming because they evaluate the correlation of ensemble member values at every pair of voxels. Here, we introduce the Local-to-Global Correlation Clustering (LoGCC) method, which leverages the ensemble's spatial structure to significantly accelerate the subdivision process. In the first, local step, LoGCC generates a partition of the ensemble's domain based solely on relationships between adjacent voxels. The second, global step recovers broader connections by leveraging the weak transitivity properties of correlation. Various heuristics can be efficiently implemented in this global step, operating on the reduced set of primitives generated in the local step. When the global perspective provided by Inclusion Depth proves unsatisfactory, LoGCC's spatial partitioning offers building blocks for a localized analysis. LoGCC is also significantly faster than previous correlation clustering methods, processing large volumetric datasets like deep ensembles within a reasonable time. Finally, similar to the Inclusion Depth methods, we developed and open-sourced a Python library to enable correlation-based analysis of scalar field ensembles in other domains.

This chapter is based on the submitted paper:

Chaves-de-Plaza, Nicolas F., R. G. Raidou, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles". Submitted for publication. 2025

The extracted ensemble information and proposed visual analysis techniques in Chapters 3-6 can readily support decision-making processes involving segmentation ensembles. Further, owing to their modularity, generality, and availability as Python packages, we envision the proposed methods can also be integrated into ensemble-supported QA workflows. To illustrate this, we provide a comprehensive example showing how the developed contour depth methodologies can be applied to segmentation ensemble analysis workflows in radiotherapy in **Chapter 7**. In **Chapter 8**, we conclude by discussing potential challenges and opportunities this endeavor poses and future research and development avenues.

REFERENCES

[1] M. Lederman. “The early history of radiotherapy: 1895–1939”. In: *International Journal of Radiation Oncology*Biology*Physics* 7.5 (1981), pp. 639–648.

[2] K. S. Huh Hyun Do. “History of Radiation Therapy Technology”. In: *pmp* 31.3 (2020), pp. 124–134.

[3] T. Hellevik and I. Martinez-Zubiaurre. “Radiotherapy and the Tumor Stroma: The Importance of Dose and Fractionation”. In: *Frontiers in Oncology* 4 (2014).

[4] H. E. Morgan and D. J. Sher. “Adaptive radiotherapy for head and neck cancer”. In: *Cancers of the Head & Neck* 5.1 (2020), p. 1.

[5] P. Blanchard, G. B. Gunn, A. Lin, R. L. Foote, N. Y. Lee, and S. J. Frank. “Proton Therapy for Head and Neck Cancers”. In: *Seminars in Radiation Oncology* 28.1 (2018), pp. 53–63.

[6] J.-J. Sonke, M. Aznar, and C. Rasch. “Adaptive Radiotherapy for Anatomical Changes”. In: *Seminars in Radiation Oncology* 29.3 (2019), pp. 245–257.

[7] J. Heukelom and C. D. Fuller. “Head and Neck Cancer Adaptive Radiation Therapy (ART): Conceptual Considerations for the Informed Clinician”. In: *Seminars in Radiation Oncology* 29.3 (2019), pp. 258–273.

[8] O. L. Green, L. E. Henke, and G. D. Hugo. “Practical Clinical Workflows for Online and Offline Adaptive Radiation Therapy”. In: *Seminars in Radiation Oncology* 29.3 (2019), pp. 219–227.

[9] M. van Herk. “Errors and margins in radiotherapy”. In: *Seminars in Radiation Oncology* 14.1 (2004), pp. 52–64.

[10] F. Albertini, M. Matter, L. Nenoff, Y. Zhang, and A. Lomax. “Online daily adaptive proton therapy”. In: *British Journal of Radiology* 93.1107 (Nov. 2019), p. 20190594.

[11] J. Bertholet, G. Anastasi, D. Noble, A. Bel, R. van Leeuwen, T. Roggen, M. Duchateau, S. Pilskog, C. Garibaldi, N. Tilly, R. García-Mollá, J. Bonaque, U. Oelfke, M. C. Aznar, and B. Heijmen. “Patterns of practice for adaptive and real-time radiation therapy (POP-ART RT) part II: Offline and online plan adaption for interfractional changes”. In: *Radiotherapy and Oncology* 153 (2020), pp. 88–96.

[12] N. F. Chaves-de-Plaza, P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, and R. van Egmond. “Report on AI-Infused Contouring Workflows for Adaptive Proton Therapy in the Head and Neck”. In: *arXiv preprint arXiv:2208.04675* (2022).

[13] M. Uffmann and C. Schaefer-Prokop. “Digital radiography: the balance between image quality and required radiation dose”. In: *European journal of radiology* 72.2 (2009), pp. 202–208.

[14] J. Lamb, M. Cao, A. Kishan, N. Agazaryan, D. H. Thomas, N. Shaverdian, Y. Yang, S. Ray, D. A. Low, A. Raldow, *et al.* “Online adaptive radiation therapy: implementation of a new process of care”. In: *Cureus* 9.8 (2017).

[15] G. Sharp, K. D. Fritscher, V. Pekar, M. Peroni, N. Shusharina, H. Veeraraghavan, and J. Yang. “Vision 20/20: Perspectives on automated image segmentation for radiotherapy”. In: *Medical Physics* 41.5 (2014), p. 050902.

[16] T. Vrtovec, D. Močnik, P. Strojan, F. Pernuš, and B. Ibragimov. “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.

[17] V. Grégoire, K. Ang, W. Budach, C. Grau, M. Hamoir, J. A. Langendijk, A. Lee, Q.-T. Le, P. Maingon, C. Nutting, B. O’Sullivan, S. V. Porceddu, and B. Lengele. “Delineation of the neck node levels for head and neck tumors: A 2013 update. DAHANCA, EORTC, HKNPCSG, NCIC CTG, NCRI, RTOG, TROG consensus guidelines”. In: *Radiotherapy and Oncology* 110.1 (2014), pp. 172–181.

[18] B. Segedin and P. Petric. “Uncertainties in target volume delineation in radiotherapy – are they relevant and what can we do about them?” In: *Radiology and Oncology* 50.3 (2016), pp. 254–262.

[19] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444.

[20] M. J. Trimpl, S. Primakov, P. Lambin, E. P. J. Stride, K. A. Vallis, and M. J. Gooding. “Beyond automatic medical image segmentation—the spectrum between fully manual and fully automatic delineation”. In: *Physics in Medicine & Biology* 67.12 (June 2022), 12TR01.

[21] C. E. Cardenas, J. Yang, B. M. Anderson, L. E. Court, and K. B. Brock. “Advances in Auto-Segmentation”. In: *Seminars in Radiation Oncology* 29.3 (2019), pp. 185–197.

[22] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Cham: Springer International Publishing, 2015, pp. 234–241.

[23] S. A. A. Kohl, B. Romera-Paredes, C. Meyer, J. D. Fauw, J. R. Ledsam, K. H. Maier-Hein, S. M. A. Eslami, D. J. Rezende, and O. Ronneberger. “A probabilistic U-net for segmentation of ambiguous images”. In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS’18. Montréal, Canada: Curran Associates Inc., 2018, pp. 6965–6975.

[24] S. A. A. Kohl, B. Romera-Paredes, K. Maier-Hein, D. J. Rezende, S. M. A. Eslami, P. Kohli, A. Zisserman, and O. Ronneberger. “A Hierarchical Probabilistic U-Net for Modeling Multi-Scale Ambiguities”. In: *ArXiv* abs/1905.13077 (2019). url: https://api.semanticscholar.org/CorpusID:170079074.

[25] M. Monteiro, L. L. Folgoc, D. C. de Castro, N. Pawlowski, B. Marques, K. Kamnitsas, M. van der Wilk, and B. Glocker. “Stochastic segmentation networks: modelling spatially correlated aleatoric uncertainty”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS ’20. Vancouver, BC, Canada: Curran Associates Inc., 2020. isbn: 9781713829546.

[26] A. Rahman, J. Valanarasu, I. Hacihaliloglu, and V. M. Patel. “Ambiguous Medical Image Segmentation Using Diffusion Models”. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, June 2023, pp. 11536–11546. doi: 10.1109/CVPR52729.2023.01110. url: https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.01110.

[27] E. Weiss and C. F. Hess. “The impact of gross tumor volume (GTV) and clinical target volume (CTV) definition on the total accuracy in radiotherapy”. In: *Strahlentherapie und Onkologie* 179.1 (2003), p. 21.

[28] D. Lin, K. A. Wahid, B. E. Nelms, R. He, M. A. Naser, S. Duke, M. V. Sherer, J. P. Christodouleas, A. S. R. Mohamed, M. Cislo, J. D. Murphy, C. D. Fuller, and E. F. Gillespie. “E pluribus unum: prospective acceptability benchmarking from the Contouring Collaborative for Consensus in Radiation Oncology crowdsourced initiative for multiobserver segmentation”. In: *Journal of Medical Imaging* 10.S1 (2023), S11903.

[29] M. Wolf, J. Krause, P. A. Carney, A. Bogart, and R. H. J. M. Kurvers. “Collective Intelligence Meets Medical Decision-Making: The Collective Outperforms the Best Radiologist”. In: *PLOS ONE* 10.8 (Aug. 2015), pp. 1–10.

[30] C. Fiorino, M. Reni, A. Bolognesi, G. M. Cattaneo, and R. Calandrino. “Intra- and inter-observer variability in contouring prostate and seminal vesicles: implications for conformal treatment planning”. In: *Radiotherapy and Oncology* 47.3 (1998), pp. 285–292.

[31] “3. Definition of Volumes”. In: *Journal of the ICRU* 14.2 (2024/10/14 2014), pp. 55–63. doi: https://doi.org/10.1093/jicru_ndx003.

[32] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Z. Chen. “Suggestive Annotation: A Deep Active Learning Framework for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2017*. Ed. by M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, and S. Duchesne. Cham: Springer International Publishing, 2017, pp. 399–407.

[33] J. Sander, B. D. de Vos, and I. Išgum. “Automatic segmentation with detection of local segmentation failures in cardiac MRI”. In: *Scientific Reports* 10.1 (2020), p. 21769.

[34] M. B. Altman, J. A. Kavanaugh, H. O. Wooten, O. L. Green, T. A. DeWees, H. Gay, W. L. Thorstad, H. Li, and S. Mutic. “A framework for automated contour quality assurance in radiation therapy including adaptive techniques”. In: *Physics in Medicine & Biology* 60.13 (June 2015), p. 5199.

[35] A. Kendall and Y. Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017.

[36] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. “Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[37] T. Pfaffelmoser and R. Westermann. “Visualization of Global Correlation Structures in Uncertain 2D Scalar Fields”. In: *Comput. Graph. Forum* 31.3pt2 (2012), pp. 1025–1034.

2

TOWARDS FAST HUMAN-CENTERED CONTOURING WORKFLOWS FOR ADAPTIVE EXTERNAL BEAM RADIOTHERAPY

As explained in Chapter 1, the segmentation process remains a bottleneck in the radiotherapy workflow. Through interviews and an observational study conducted at Leiden University Medical Center and Holland Proton Therapy Center in the Netherlands, this chapter identifies the main challenges and opportunities for accelerating the quality assessment of segmentations in adaptive therapies. We uncovered three context-dependent variables that influence segmentation performance: usable additional information, applicable domain-specific knowledge, and the available editing capabilities of contouring software. These variables are instrumental in enabling clinicians to focus their attention on critical areas and dynamically adjust the required level of precision depending on clinical needs. By shedding light on these workflows, this chapter provides a foundation for ensemble-driven techniques discussed in subsequent chapters, offering a framework to align computational tools with clinicians' decision-making processes.

This chapter is based on the published paper: Chaves-de-Plaza, N. F., Mody, P., Hildebrandt, K., Staring, M., Astreinidou, E., de Ridder, M., de Ridder, H. & van Egmond, R. "Towards fast human-centred contouring workflows for adaptive external beam radiotherapy". In: Proceedings of the Human Factors and Ergonomics Society Europe. 2022, pp. 111-31

2.1. INTRODUCTION

External Beam Radiotherapy (EBRT) is the most common form of RT and has become one of humanity's main tools against cancer, together with surgery and systemic treatment. In EBRT, ionizing radiation is directed at the patient's tumor to destroy the malignant cells. Over the last decades, significant technological improvements have been made in treatment planning and delivery, which increased the precision of EBRT. For instance, proton beam therapy (PT) can harness the ability of protons to deposit all their energy at a specific spot [1, 2]. This capability permits PT more precisely shape the radiation dose to the tumor, minimizing the dose to the surrounding healthy tissue and reducing side effects [3–6].

Harnessing the precision increase of dose delivery technology requires adapting the patient's treatment plan to the anatomy of the day. Figure 2.1 presents the general workflow of this treatment paradigm known as adaptive EBRT. Adaptive EBRT imposes severe time constraints on online treatment planning processes (orange boxes in Figure 2.1) because longer within fraction times can lead to new anatomical changes, offsetting the value of the adaptation. Also, an increase in the footprint of treatment planning processes would reduce patient throughput, compromising the viability of adaptive EBRT.

The present study investigates the challenges that the contouring process poses to the implementation of adaptive EBRT. Despite the availability of auto-contouring technologies, contouring remains human-centered because clinicians need to perform an extensive quality assessment of the generated delineations to ensure that they do not contain inaccuracies [7–10]. Therefore, to reduce the footprint of the contouring process, it is necessary to understand human factors that impact its duration. This study extends prior works in two ways. First, it focuses on the time dimension of contouring performance, uncovering factors that influence it. Traditionally, researchers have directed their attention to analyzing the effect of different image modalities, guidelines, contouring software, and experience on output-based performance metrics like accuracy and inter-observer contouring variability [11–15]. This focus makes sense considering the influence that these metrics have on patient safety [16, 17]. Nevertheless, factors that affect time can also impact accuracy, motivating the need to study them. On the one hand, other things equal, accuracy degrades in time-constrained scenarios [18, 19]. On the other, if clinicians perform demanding tasks for extended periods, they can become fatigued and lose situation awareness, which will also impact accuracy [20, 21]. Second, this work studies the contouring process in its clinical context. Prior works have investigated the effect of input devices and user interfaces on contouring time using experiments in highly controlled environments [13, 22, 23]. These studies' findings hold for the general contouring case. Nevertheless, this needs not to be the case in the time-constrained phase of adaptive EBRT (orange boxes in Figure 2.1). This study follows a qualitative context-driven approach to uncover factors that affect contouring performance in adaptive EBRT and discusses potential context-aware strategies to mitigate them. Adopting an ecological approach to researching human factors that affect contouring performance can help designing representative experiments and evaluations for contouring in time-critical scenarios [24]. Furthermore, the findings from this study represent the initial step of methodologies like Ecological Interface Design, which aims to develop systems that promote adaptive performance [25]. To summarize, the present study investigates factors that affect the duration of the contouring process and discusses potential mitigation strategies. It complements and extends prior studies that analyzed human factors of contouring performance [26, 27], providing an updated account of the process workflow in the time-critical context of adaptive EBRT. Finally, the present study contributes to the state-of-the-art of clinical contouring workflows in adaptive EBRT in two ways:

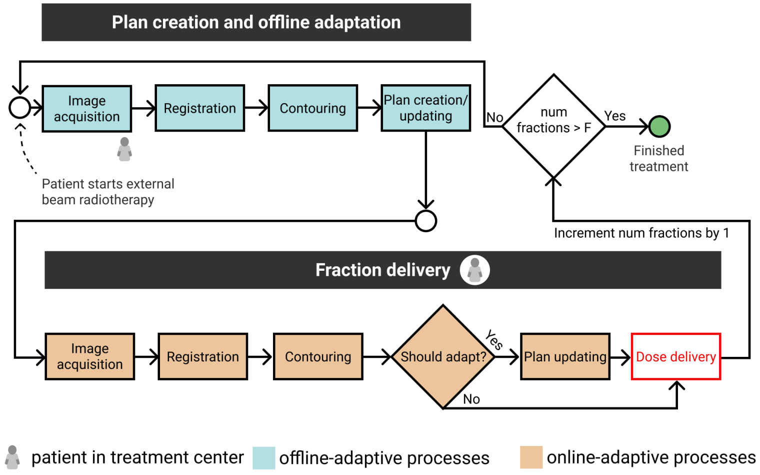


Figure 2.1.: Schematic of external beam radiotherapy (EBRT) dose delivery pipeline. Each box corresponds to one process, and the diamonds to decisions in the workflow. The goal is to deliver the prescribed dose to the patient (red box) in F fractions spread over several days. Adaptive strategies help mitigate dose deviations due to changes in the patient's anatomy during the treatment. Adaptation can be online within a fraction (orange boxes) or offline between fractions (blue boxes).

- It reports the results of an observational study in two cancer treatment centers in the Netherlands. The study of the Contouring Workflow provided a situated account of the current contouring workflows in the context of adaptive EBRT, together with factors that can affect its performance.

- It discusses acceleration strategies based on the context of adaptive EBRT that tool developers and clinicians can leverage to adapt the contouring workflow to time-constrained scenarios.

2.2. THE CONTOURING ACTIVITY

An exploratory literature review was performed to establish baseline knowledge about the contouring activity and its role in adaptive therapies. The query used for the search (Scopus, PubMed, and Google Scholar) included the keywords: adaptive, adaptation, proton therapy, radiotherapy, contouring, automatic, semi-automatic, workflow, and head-and-neck. The latter term was relevant since the study's participants (next section) were specialists in this region. The search yielded around 50 articles with publishing years ranging between 2008 and 2021. As Figure 2.2 depicts, the main inputs of the contouring activity are 3D images (stacks of hundred of 2D images) that describe the patient anatomy. Among these, there is an image to contour, usually a Computerized Tomography (CT), and supporting information such as previous contours of the patient and other image modalities such as Magnetic Resonance Imaging (MRI) and Positron Emission Technology CT (PET-CT). Using available information, contouring consists of drawing the boundaries of anatomical structures relevant to the patient's cancer in the image to contour. The two main anatomical groups are the target volumes (TVs), which correspond to areas affected by tumoral cells, and the organs at risk (OARs), which correspond to healthy tissue.

As the right panel of Figure 2.2 indicates, the goal of the contouring activity is to produce contours suitable for creating or updating the patient treatment plan and assessing its quality. Several actors participate in this workflow in the clinic, distributing contouring tasks based on the anatomical structures' groups. In general, radiotherapy technologists (RTTs) start by delineating the OARs. After this, the radiation oncologists (ROs), who are directly responsible for the patient's outcome, assess the quality of the OARs contours and draw the boundaries of the TVs, the structures with the highest priority. The study described in the next section was designed based on this understanding of the contouring activity.

2.3. STUDY OF THE CONTOURING WORKflOW

A study of the contouring workflow was conducted to identify characteristics of adaptive EBRT affecting contouring performance and to identify context-dependent strategies that tool developers can leverage to improve it. The following subsections detail the study's design and describe the methodology used for analyzing the resulting data.

2.3.1. STUDY DESIGN

PARTICIPANTS

Two radiation oncologists (RO) and two radiotherapy technologists (RTT) from two cancer treatment centers in the Netherlands specializing in the head-and-neck area joined the study. Table 2.1 summarizes the participants' information. One of the institutes, the Leiden University Medical Center (LUMC), offers photon-based volumetric modulated arc therapy (VMAT) treatments. The second, the Holland Proton Therapy Center (HollandPTC), offers proton therapy (PT). Despite the differences in dose delivery technology, both institutions have a similar workflow, performing offline adaptations. The latter means that the patient's treatment plan is updated sparsely during treatment (entails re-executing blue boxes in Figure 2.1). The Institutional Review Board at the Delft University of Technology approved this research. Each participant provided informed consent to be part of the study.

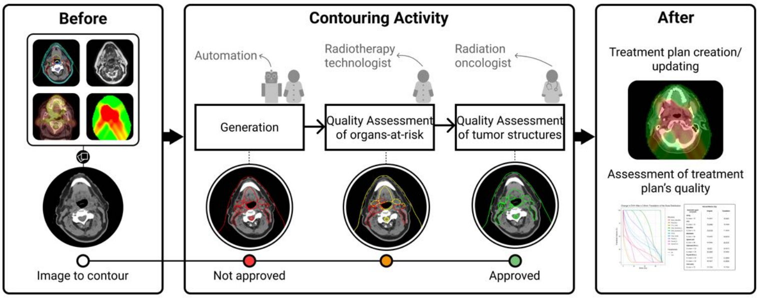

Before
Image to contour
Contouring Activity
Automation
Radiotherapy technologist
Radiation oncologist
Generation
Quality Assessment of organs-at-risk
Quality Assessment of tumor structures
Not approved
Approved
After
Treatment plan creation/ updating
Assessment of treatment plan's quality

Figure 2.2.: Components of the contouring activity. The inputs (left) are the image to contour and, optionally, other three-dimensional datasets like MRI and PET-CT scans and dose distribution volumes. The contouring activity has two main processes that several actors perform: generation of contours and its quality assessment. After approving the contours, clinicians can use them to create/update the patient's treatment plan and assess its quality.

PROCEDURE

The study had three sessions. The first one, a one-hour-long semistructured interview, permitted establishing rapport with the participants and validated the initial understanding of the EBRT workflow. In the second and third sessions, the participants performed their contouring duties while being recorded. As Table 2.1 shows, these meetings lasted between one and two hours, depending on the participants' time. In the second session, clinicians performed initial contouring. The third focused on adaptive contouring, where clinicians perform a quality assessment of automatically generated contours. Given the limited clinicians' time to participate, they contoured a subset of anatomical including the tumors and organs close to them that could affect the patient outcome.

Table 2.1.: Participants of the qualitative sessions. Two radiation oncologists (RO) and two radiotherapy technologists (RTT) from two institutions in the Netherlands participated. In some cases, due to their tight schedules, they could not attend all the sessions.

ID	Institution	Role	Session	Time (hours)
P1	LUMC	RO	1,2,3	5
P2	LUMC	RTT	2,3	2
P3	HollandPTC	RO	1,2	3
P4	HollandPTC	RTT	1,2,3	5

MATERIALS

For the observational sessions, clinicians at each center had access to the data of two previously treated head and neck patients. Each patient file included initial treatment planning data such as CT, PET-CT, and MRI scans and daily images such as CBCT and CT, relevant for sessions 2 and 3, respectively. For session 3, starting delineations could have been generated by another clinician or automated methods like deformable or rigid registration and deep learning-based contouring. For inspecting and editing the contours, clinicians used their routine software.

2.3.2. DATA ANALYSIS

The recordings of the three sessions were transcribed and analyzed using Thematic Analysis [28]. The coding process was bottom-up, first labeling patterns in the transcripts and then grouping the resulting fine-grained codes into coarser ones based on their similarity. Table 2.2 displays the underlying coarser codes, the resulting themes, and sample data excerpts. The screen recordings of sessions 2 and 3 were also relevant as they showcased the way clinicians interact with the user interface during the contouring process. The interactions were mapped onto a timeline like the one that Figure 2.4 depicts. For the y-axis, the authors drew inspiration from the literature on contouring tasks [29] but grouped them into four categories to simplify the coding process and the analysis. These are direct and indirect manipulation, navigation, and non-contouring interactions.

2.4. INITIAL CONTOURING

2.4.1. RESULTS

Initial contouring (IC) occurs when executing the plan creation and offline adaptation process in Figure 2.1 for the first time. At LUMC and HollandPTC, initial contouring (IC) takes two to six hours for head-and-neck (HN) cancers, requiring delineating more than twenty structures. The following paragraphs group the observations about the IC workflow into three characteristics, finishing with a discussion on how these can affect contouring performance.

USABLE ADDITIONAL INFORMATION

At IC, no pre-existing contours of the patients exist, given that this process occurs after they have started treatment. Instead, clinicians use information from multiple image modalities acquired beforehand. The main image modality in radiotherapy, CT, usually does not provide enough boundary information when the contrast between adjacent tissues is not enough or when there is noise or artifacts in the image acquisition process. In these cases, clinicians rely on Magnetic Resonance Imaging (MRI) and Positron Emission Technology-CT (PET-CT) scans, acquired for most patients at HollandPTC and LUMC. As Figure 2.3 shows, MRI helps differentiate soft tissue structures: “MRI makes it easier for us to delineate the parotid glands because you can see them very good at an MRI.”. For PET-CT, this modality permits clinicians to locate tumors and estimate their boundaries with higher precision: “We actually scan all of our head and neck patients [with PET-CT] because it makes our delineations so much accurate, so that is now standard.” [P1].

In practice, clinicians align additional images to the CT before using them for contouring. This process, known as image registration, can take several minutes per image pair and requires the clinician’s intervention to verify the alignment’s quality. Registering the images allows clinicians to scroll through them in parallel using the contouring software, enabling direct comparison of the structures in both scans.

Table 2.2.: The first column presents the themes that emerged during the Thematic Analysis of the transcripts of the semi-structured interviews and observational sessions of the Study of the Contouring Workflow. The second column presents the coarser codes obtained after several grouping iterations finer ones. Lastly, the third column displays, for each theme, a representative example from the transcribed data.

Theme	Codes	Example
Adaptive contouring context	Clinical workflow, standardization, physical and clinical artifacts, training, institution specific considerations, EBRT technology	“Now it takes one day to do the whole plan. So, we have to make a new calculation and it has to go into the the LINAC so it has to get another check.” [P2]
Structure priority and effect of innacuracies on patient’s treatment	Anatomical knowledge, downstream effects, characteristics of different anatomical structures, clinical priorities, tumor-related considerations	“I guess if it’s an inner region where for instance the cheek region here. Those are minor [edits], but if we see this region where you have the parotid gland. There it could influence dose to the OARs quite significantly. So there. Then I would say it’s a major [edit].” [P1]
Dealing with uncertain regions in the image-to-contour	Anatomical knowledge, image modalities, papers and guidelines, information required for certainty	“With the nasopharyngeal cancers, then I will take an MRI and then I will draw on the MRI. So, then I know exactly where the brainstem is.” [P4]
Editing capabilities of contouring software	Characteristics of contouring software, experience with the tools, use of automation	“It seems to me that it’s a model based one [automatically generated contour] because the model based one always has trouble here at the head of the mandible at the joint.” [P3]
Distribution of labor and clinicians experience	Experience with the contouring task, collaboration, task distribution, protocols	“When an RTT does it [a contour]? Sometimes it’s very nice and when a not so experienced RTT does it it’s not a very good delineation and then it costs me either a lot of time to adjust every slice or I just start again and that’s most of the time.” [P3]

APPLICABLE DOMAIN-SPECIFIC KNOWLEDGE

In some cases, the information in the images is not enough. At IC, this happens when MRI and PET-CT scans are not available and moreover there are no pre-existing contours of the patients (they just started the treatment). In these cases, clinicians rely on domain-specific knowledge they access in two ways. First, they leverage guidelines [30] and atlases that describe and indicate what the contours should look like, respectively. Second, they draw on their experience. Experienced clinicians know what areas can be challenging to delineate given the available data. They use this domain-specific anatomical knowledge to direct their attention and estimate contours over unclear image boundaries. An example of this dynamic occurs when the radiation oncologists (ROs) review the delineations created by the radiotherapy technologists (RTTs): "We [ROs] think that it [delineating the swallowing muscles] is too hard for RTTs, need quite a bit of anatomical knowledge to know where they are exactly. And in this case, this patient doesn't have a very big tumor in the throat, but most of the time patients have

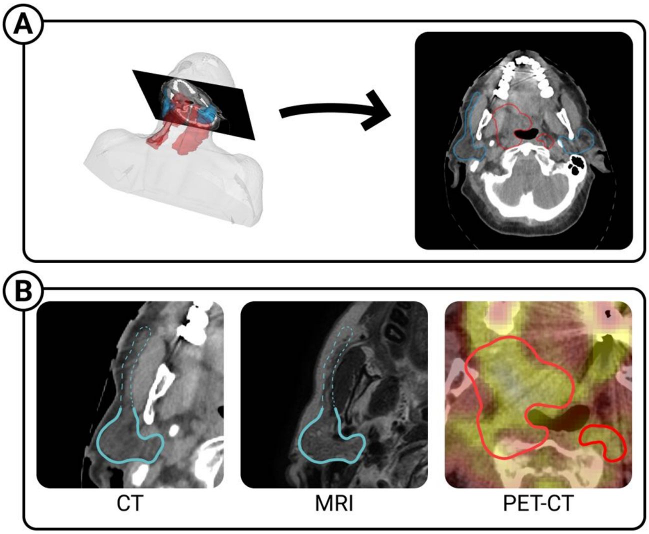


Figure 2.3.: Available information available at contouring. The central input is the image to contour which, as panel A depicts, is a three-dimensional image made from several 2D slices. Other three-dimensional images available at the surveyed centers are magnetic resonance imaging (MRI) and positron imaging technology CT (PET-CT) scans. As panel B shows, MRI helps differentiate soft tissue, and PET-CT aids in detecting and delineating tumors.

quite a big tumor here. And you can't see the swallowing muscles that good. So, then you need to know exactly where they run from to delineate them." [P1].

EDITING CAPABILITIES OF CONTOURING SOFTWARE

In practice, at IC, clinicians create the contours from scratch. As the timeline on the top section of Figure 2.4 depicts, this entails starting with an empty delineation and gradually building the contours through a series of interactions. At the surveyed institutions, clinicians favored a semi-automatic workflow, which consisted of two phases. First, they generated initial contours using the between-slice interpolation tool. This tool requires clinicians to manually delineate a subset of the slices spanning the structure, after which the rest of the structure's contours will be interpolated (this autocompletion corresponds to the indirect editing interaction around the second eighty in Figure 2.4). Finally, revert to the manual brush tool to correct inaccuracies. As the timeline shows, the generation of contours takes more time than the refinement, and clinicians spend most of the time directly editing the delineations with the brush.

2.4.2. DISCUSSION

Clinicians use contours produced at IC to create the patient's treatment plan. Therefore, they seek maximal accuracy, often at the expense of longer task durations. The three characteristics of the IC context described before affect contouring time in several ways. First, extra image modalities reduce the task difficulty, which can result in reduced dwelling times to determine where the contour should go. Nevertheless, additional images need to be registered to the main one, a time-consuming process that could offset the performance benefits gains that the process offers. Second, domain-specific knowledge can reduce the extent of the contouring task by letting clinicians direct their attention to where it is needed. Yet, following the accuracy directive, they still must go through the whole volume to ensure no inaccuracy remains. Finally, the semi-automatic between slice interpolation tool spares clinicians from needing to edit several slices but still requires significant manual effort to initialize the method.

2.5. ADAPTIVE CONTOURING

2.5.1. RESULTS

LUMC and HollandPTC implement an offline-adaptive dose delivery pipeline, which entails updating the treatment plan several times during treatment by repeating the plan creation and offline adaptation process between fractions. Adaptive contouring (AC) occurs in this setting and differs from initial contouring (IC) in that the time is more critical and the resources scarcer. At the surveyed institutions, AC takes one to two hours for head and neck cancer patients. Like the previous section, the following paragraphs detail the AC context and discuss how it affects the process' performance.

USABLE ADDITIONAL INFORMATION

In contrast with IC, at AC, no extra images of the patient are acquired. Therefore, clinicians have access to the image to contour, a CT at LUMC and HollandPTC, the images acquired for IC, and the approved IC

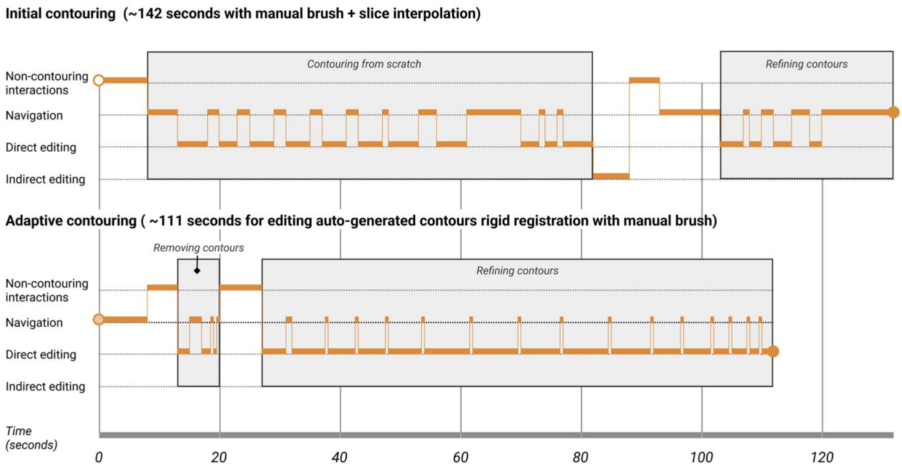


Figure 2.4.: Interaction timelines for initial and adaptive contouring. In both cases, a radiotherapy technologist from LUMC (P2 in Table 2.1), delineated the right submandibular gland of a head and neck cancer patient. The x-axis encodes time, and the y-axis differentiates the principal interaction categories. Non-contouring interactions correspond to changes in the interface that do not affect the contours, like changing the layout or visualization parameters. Navigation refers to changing the current slice of the image to contour. Finally, direct and indirect manipulations entail altering the delineations in the 2D slice or through a button in the menu, respectively. Note how initial contouring starts from scratch (empty circle) while adaptive contouring starts with pre-generated delineations (partially filled circle).

contours. In practice, clinicians only use the latter and do so in two ways. First, because IC contours document all the clinical decisions made for the current patient, they use them as a patient-specific atlas to resolve complex contouring tasks. Regarding having an atlas for contouring, P4 mentioned that “it’s always nice to have it [the atlas] like a verification. Because the brainstem isn’t that difficult, but like if you have the swallowing muscles or something, that’s really something. If you have the atlas side by side, it really can come in handy.” [P4] Second, clinicians use approved IC contours to create an initial segmentation. For this, they align, or register, the IC and AC images and then “propagate” the contours from the former to the latter.

APPLICABLE DOMAIN-SPECIFIC KNOWLEDGE

In addition to general anatomical knowledge, at AC, clinicians use knowledge about dosimetry and the patient tumor to structure and guide the contouring process. On the one hand, it can help them direct their attention to critical areas. On the other, it lets them modulate the contouring based on the structure’s relevance to the patient’s treatment plan. For instance, P2 mentioned that while some contours require maximal attention and precision: “. . . with this type of organs, as with all the nervical organs, as in optical nerves and brain stem and spinal cord, when it’s critical, so when the PTV is nearby, then it’s very important that we draw this very precise.” Others accept rougher contours as they will not significantly impact the patient’s outcome: “this submandibular gland, it gets too much dose, so it won’t work. After irradiation, this one is gone. So, at that point, we can decide to delineate, but it isn’t, it’s OK if it isn’t quite perfect.”

EDITING CAPABILITIES OF CONTOURING SOFTWARE

As mentioned before, clinicians do not start delineating from scratch at AC. Instead, they generate a starting point by propagating the contours from the initial scan to the current one. Therefore, the goal at AC is to perform a quality assessment (QA) of these delineations. The timeline in the bottom section of Figure 2.4 exemplifies the series of interactions that clinicians usually perform during the QA process. In the timeline, it is possible to see how starting from partial delineations, they reach the final ones after a series of relatively long direct editing interactions interleaved with brief navigation operation ones. Between slice interpolation, the tool clinicians use for contouring from scratch does not work for contour refinement. Therefore, for extensive errors across multiple slices like the one Figure 2.5 depicts, clinicians face two options. Either manually fix the contour on every slide or delete the delineation and re-do it from scratch using between-slice interpolation.

2.5.2. DISCUSSION

While clinicians use IC contours for creating the treatment plan, they use AC contours to update the plan. For this reason, at this stage, their primary concern therefore seemed to be to faithfully translate IC contours to the current patient anatomy. The identified contextual characteristics affect AC performance in several ways. First, having information about the role that each structure plays in the patient's treatment helps direct clinicians' attention to delineations that can affect the patient outcome. A potential pitfall of the current prioritization approach is that it is purely heuristic and based on clinicians' experience instead of available information such as the planned dose. Second, by using IC-approved contours, clinicians can reduce the time for analyzing and editing complex or large regions by propagating them via registration. Nevertheless, same as with other image modalities at IC, the time it takes to perform the registration might offset the time gains.

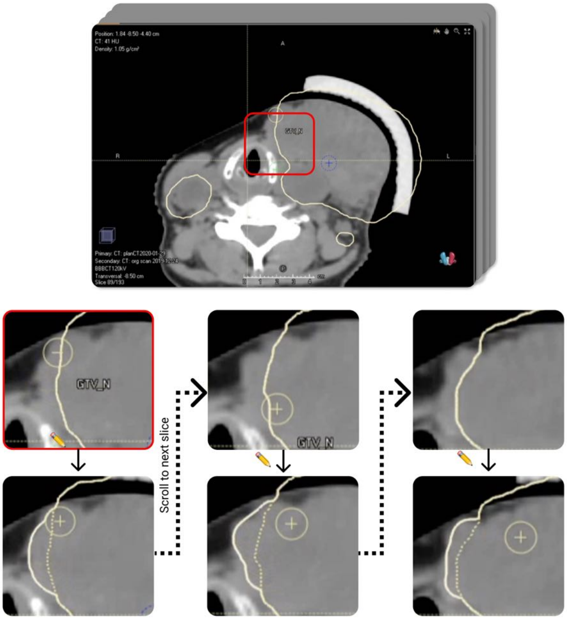


Figure 2.5.: Editing faulty delineations often entails redundant interactions. The top image presents an inaccurate auto-generated contour of a tumoral structure. As can be observed, the internal side of the contour fails to include the whole structure, which causes an error that spans three slices. The images below present the sequence of steps that P1 followed to amend the inaccuracy.

Finally, although contouring is overall faster at AC due to the contours being pre-generated, there is no tool to efficiently perform QA, requiring clinicians to invest significant manual effort.

2.6. DISCUSSION

The Study of the Contouring Workflow provided an understanding of several characteristics that affect contouring duration in adaptive EBRT. This section takes these observations as input and lays down several ways of accelerating the adaptive contouring activity, which is increasingly time-pressured due to clinics implementing more responsive adaptative workflows. The discussion differentiates between the inspection, navigation, and editing tasks, which account for most of the delineation time. Figure 2.6 summarizes the study's findings and the resulting context-dependent acceleration strategies.

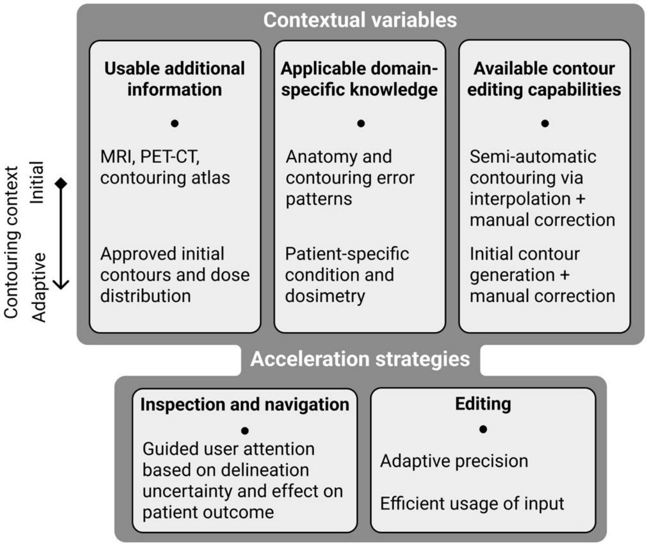

Contextual variables
Usable additional information
Applicable domain-specific knowledge
Available contour editing capabilities
MRI, PET-CT, contouring atlas
Anatomy and contouring error patterns
Semi-automatic contouring via interpolation + manual correction
Approved initial contours and dose distribution
Patient-specific condition and dosimetry
Initial contour generation + manual correction
Contouring context
Initial
Adaptive
Acceleration strategies
Inspection and navigation
Guided user attention based on delineation uncertainty and effect on patient outcome
Editing
Adaptive precision
Efficient usage of input

Figure 2.6.: Schematic of the approach that the present study followed. First, it identified three variables that influence contouring performance and described their roles in the initial and adaptive contouring contexts. These variables were then mapped to strategies for accelerating the inspection, navigation, and editing tasks.

2.6.1. INSPECTION AND NAVIGATION

In adaptive contouring, clinicians prioritized inspection of tumor contours because an error could result in overexposure of surrounding organs to radiation or, worse, in underexposure of the cancerous tissue [31]. This observation suggests that patient-specific treatment-level information provides a valuable signal to define the contouring priority of anatomical structures. Heuristics based on dose information allow clinicians to decide faster [32]. Nevertheless, problems like cognitive bias, loss of situation awareness, or varying levels of experience can introduce inconsistencies in a heuristic-based contouring process, which could risk patient safety [33, 34]. Protocols and checklists could be implemented to enable effective heuristics usage while mitigating their pitfalls [35–37]. These could be based on metrics like Normal Tissue Complication Probability (NTCP) that have been shown to affect the patient outcome [12]. Figure 2.7 presents an example of prioritization based on the local characteristics of the dose distribution. As can be observed, while a potential inaccuracy in the tumor delineation has a high priority, errors in the parotid glands are less urgent due to their lower impact on the patient's treatment. Before prioritizing errors, clinicians need to detect them. Several methods have been proposed in the literature for assisting this task. They vary in the information and the mechanism used to perform the search. As for the former, it is possible to compute shape [38, 39] and image or appearance-related [40] characteristics of the contours, e.g. the surface area or the intensity histogram, respectively. Another possible indicator of the contours' quality is their uncertainty or variability, which can come from historical patient data [41], the auto-contouring algorithm [42, 43], or directly from the image-to-contour [44]. After gathering all these sources of information, available techniques identify potential errors in two ways. Firstly, by letting a classifier automatically find data-based rules for separating inaccurate from the accurate regions [45–51]. Secondly, they delegate the search task to the users, presenting them with the traditional two-dimensional image and contour slices together with informative overlays such as uncertainty iso-lines [52, 53] and contour boxplots [54]. These two-dimensional visualizations have been augmented by adding three-dimensional views [55, 56] and letting the user interact with the data by filtering and sorting mechanisms [57, 58]. Two challenges that existing error detection tools face are maintaining users' trust in the system and lowering the cognitive load they impose. As to the former, a system failing to spot inaccuracies that affect the patient's treatment (false negatives) would erode the users' trust [59, 60]. This might explain the limited adoption of automatic error detection systems in clinical practice. Regarding cognitive load, abrupt context changes when guiding clinicians' attention to different parts of the 3D image can build up fatigue, potentially leading to errors like classifying a true positive the system suggested as a false positive [61, 62]. Visualization methods like 3D views complementing attention guidance mechanisms could help mitigate this issue.

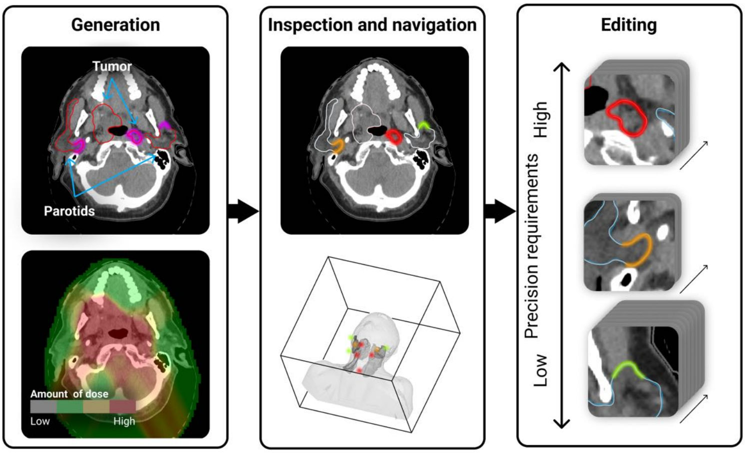


Figure 2.7.: Components for accelerating the inspection, navigation, and editing tasks. The first step (leftmost column) is to generate the contours and gather extra information like delineation variability and the dose distribution. Based on these sources, potential errors can be flagged and categorized depending on their effect on the patient outcome. In the example, an error in the tumor's delineations was flagged as high priority (red) because it can significantly change the treatment plan. As for the parotid glands, the orange inaccuracy is in a region where the dose distribution varies more quickly than in the case of the green one. Therefore, subsequent processes (like treatment plan updating) that rely on the orange contours could be more sensitive to changes in these contours.

2.6.2. EDITING

Currently, clinicians use mostly manual tools when fixing an inaccuracy. For errors that occupy a large portion of the volume, like the example in Figure 2.5, this often means that the user will perform similar edits across slices. Existing semi-automatic interactive contouring techniques mitigate this issue by extrapolating rough feedback provided by the clinician. Their general workflow consists of two steps. First, the clinician provides a rough indication of the change to be made or the area to update via coarse inputs such as scribbles, points, or a bounding box.

Based on this input, the algorithm proceeds to update the segmentation. Traditionally Markov Random Field-based algorithms are being used [63, 64]. Recently, deep learning-based implementations have appeared that offer more sophisticated suggestions based on the clinician's input [65–67]. The adoption of these semi-automatic interactive editing tools in the clinic remains challenging. Based on discussions with clinicians, the reason for their resistance to these interactive editing tools seems to be that they perceive scribbles as a blunt tool for communicating to the algorithm what they want. Therefore, more research is needed to determine which type of input mechanism the clinicians prefer and how the algorithm should respond [68, 69]. For instance, do they prefer coarse inputs like scribbles? Or would they be more comfortable with high precision inputs such as selecting a contour from an ensemble of candidates [70]? With editing being the most time-consuming QA operation, obtaining a synergy between humans and AI is paramount.

2.7. LIMITATIONS AND FUTURE WORK

A limitation of this work is the reduced number of treatment centers and clinicians surveyed in the study, which might have led to weighting heavily on custom institutional practices and personal preferences. As a promising solution, questionnaires like the one reported in [71] could be prepared to validate the conclusions with a larger pool of participants. Another limitation is the qualitative nature of the timelines used to illustrate the dynamics between the clinicians and the contouring software. In further studies, we plan to use keystroke logging software to include more fine-grained actions and more accurate timings. The latter would be especially valuable for comparing different segmentation tools. In terms of future work, we will translate the findings of this study into a practical human-centered contouring protocol that clinicians can adapt to their institution-specific adaptive EBRT capabilities and constraints. In addition to the clinician-level considerations that the present article considered, such protocol will also account for team dynamics, which also emerged as a performance factor in the surveyed institutions.

2.8. CONCLUSION

This study characterized the contouring workflows in adaptive EBRT. An observational study at two treatment centers in the Netherlands revealed several context-dependent characteristics that influence delineation performance. Based on these observations, strategies for accelerating inspection, navigation, and editing tasks were discussed. By applying these when developing and commissioning tools, tool builders and clinicians can decrease the delineation time and thus increase the suitability of this process for time-critical therapies like online-adaptive EBRT.

REFERENCES

[1] W. D. Newhauser and R. Zhang. “The physics of proton therapy”. In: *Physics in Medicine & Biology* 60.8 (2015), R155.

[2] R. R. Wilson. “Radiological Use of Fast Protons”. In: *Radiology* 47.5 (1946), pp. 487–491.

[3] J. A. Langendijk, P. Lambin, D. De Ruysscher, J. Widder, M. Bos, and M. Verheij. “Selection of patients for radiotherapy with protons aiming at reduction of side effects: The model-based approach”. In: *Radiotherapy and Oncology* 107.3 (2013), pp. 267–273.

[4] J. Lundkvist, M. Ekman, S. Rehn Ericsson, B. Jönsson, and B. Glimelius. “Proton therapy of cancer: Potential clinical advantages and cost-effectiveness”. In: *Acta Oncologica* 44.8 (2005), pp. 850–861.

[5] I. Simone Charles B., D. Ly, T. D. Dan, J. Ondos, H. Ning, A. Belard, J. O’Connell, R. W. Miller, and N. L. Simone. “Comparison of intensity-modulated radiotherapy, adaptive radiotherapy, proton radiotherapy, and adaptive proton radiotherapy for treatment of locally advanced head and neck cancer”. In: *Radiotherapy and Oncology* 101.3 (2011), pp. 376–382.

[6] H. Thomas and B. Timmermann. “Paediatric proton therapy”. In: *British Journal of Radiology* 93.1107 (Sept. 2019), p. 20190601.

[7] C. E. Cardenas, J. Yang, B. M. Anderson, L. E. Court, and K. B. Brock. “Advances in Auto-Segmentation”. In: *Seminars in Radiation Oncology* 29.3 (2019), pp. 185–197.

[8] S. Nikolov, S. Blackwell, A. Zverovitch, R. Mendes, M. Livne, J. De Fauw, Y. Patel, C. Meyer, H. Askham, B. Romera-Paredes, C. Kelly, A. Karthikesalingam, C. Chu, D. Carnell, C. Boon, D. D’Souza, S. A. Moinuddin, B. Garie, Y. McQuinlan, S. Ireland, K. Hampton, K. Fuller, H. Montgomery, G. Rees, M. Suleyman, T. Back, C. O. Hughes, J. R. Ledsam, and O. Ronneberger. “Clinically Applicable Segmentation of Head and Neck Anatomy for Radiotherapy: Deep Learning Algorithm Development and Validation Study”. In: *J Med Internet Res* 23.7 (2021), e26151.

[9] L. V. van Dijk, L. Van den Bosch, P. Aljabar, D. Peressutti, S. Both, R. J. H. M. Steenbakkers, J. A. Langendijk, M. J. Gooding, and C. L. Brouwer. “Improving automatic delineation for head and neck organs at risk by Deep Learning Contouring”. In: *Radiotherapy and Oncology* 142 (2020), pp. 115–123.

[10] L. Vandewinckele, M. Claessens, A. Dinkla, C. Brouwer, W. Crijns, D. Verellen, and W. van Elmpt. “Overview of artificial intelligence-based applications in radiotherapy: Recommendations for implementation and quality assurance”. In: *Radiotherapy and Oncology* 153 (2020), pp. 55–66.

[11] J. E. Bekelman, S. Wolden, and N. Lee. “Head-and-Neck Target Delineation Among Radiation Oncology Residents After a Teaching Intervention: A Prospective, Blinded Pilot Study”. In: *International Journal of Radiation Oncology, Biology, Physics* 73.2 (2009), pp. 416–423.

[12] C. L. Brouwer, R. J. H. M. Steenbakkers, E. Gort, M. E. Kamphuis, H. P. van der Laan, A. A. van’t Veld, N. M. Sijtsema, and J. A. Langendijk. “Differences in delineation guidelines for head and neck cancer result in inconsistent reported dose and corresponding NTCP”. In: *Radiotherapy and Oncology* 111.1 (2014), pp. 148–152.

[13] R. J. Steenbakkers, J. C. Duppen, I. Fitton, K. E. Deurloo, L. Zijp, A. L. Uitterhoeve, P. T. Rodrigus, G. W. Kramer, J. Bussink, K. D. Jaeger, J. S. Belderbos, A. A. Hart, P. J. Nowak, M. van Herk, and C. R. Rasch. “Observer variation in target volume delineation of lung cancer related to radiation oncologist–computer interaction: A ‘Big Brother’ evaluation”. In: *Radiotherapy and Oncology* 77.2 (2005), pp. 182–190.

[14] R. J. Steenbakkers, J. C. Duppen, I. Fitton, K. E. Deurloo, L. J. Zijp, E. F. Comans, A. L. Uitterhoeve, P. T. Rodrigus, G. W. Kramer, J. Bussink, K. De Jaeger, J. S. Belderbos, P. J. Nowak, M. van Herk, and C. R. Rasch. "Reduction of observer variation using matched CT-PET for lung cancer delineation: A three-dimensional analysis". In: *International Journal of Radiation Oncology*Biology*Physics* 64.2 (2006), pp. 435–448.

[15] S. K. Vinod, M. Min, M. G. Jameson, and L. C. Holloway. "A review of interventions to reduce inter-observer variability in volume delineation in radiation oncology". In: *Journal of Medical Imaging and Radiation Oncology* 60.3 (2016), pp. 393–406.

[16] B.-T. Karsh, R. Holden, S. Alper, and C. K. Or. "A human factors engineering paradigm for patient safety: Designing to support the performance of the healthcare professional". In: *Quality & safety in health care* 15 Suppl 1 (Jan. 2007), pp. i59–65. doi: 10.1136/qshc.2005.015974.

[17] C. Njeh. "Tumor delineation: The weakest link in the search for accuracy in radiotherapy". In: *Journal of medical physics* 33.4 (2008), pp. 136–140.

[18] M. Chignell, T. Tong, S. Mizobuchi, and W. Walmsley. "Combining Speed and Accuracy into a Global Measure of Performance". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 58.1 (2014), pp. 1442–1446.

[19] R. W. Pew. "The speed-accuracy operating characteristic". In: *Acta Psychologica* 30 (1969), pp. 16–26.

[20] M. R. Endsley. "SITUATION AWARENESS". In: John Wiley & Sons, Ltd, 2021. Chap. 17, pp. 434–455.

[21] S. B. Evans, D. Cain, A. Kapur, D. Brown, and T. Pawlicki. "Why Smart Oncology Clinicians do Dumb Things: A Review of Cognitive Bias in Radiation Oncology". In: *Practical Radiation Oncology* 9.4 (2019), e347–e355.

[22] M.-I. T. D. in Oncology Group. "Human–Computer Interaction in Radiotherapy Target Volume Delineation: A Prospective, Multi-institutional Comparison of User Input Devices". In: *Journal of Digital Imaging* 24.5 (2011), pp. 794–803.

[23] A. Ramkumar. "HCI in interactive segmentation: Human-computer interaction in interactive segmentation of CT images for radiotherapy". English. Dissertation (TU Delft). Delft University of Technology, 2017. isbn: 978-94-92516-47-3. doi: 10.4233/uuid:0f0259f1-0c33-442f-b851-86a846e736fc.

[24] J. M. Flach, P. A. Hancock, J. Caird, and K. J. Vicente. *Global perspectives on the ecology of human-machine systems*. CRC Press, 2018.

[25] K. J. Vicente. "Ecological Interface Design: Progress and Challenges". In: *Human Factors* 44.1 (2002), pp. 62–78.

[26] A. Aselmaa, R. Goossens, B. Rowland, A. Laprie, W. Song, and A. Freudenthal. "Medical Factors of Brain Tumor Delineation in Radiotherapy for Software Design". In: *Proceedings of the 5th International Conference on Applied Human Factors and Ergonomics*. July 2014. doi: 10.54941/ahfe100531.

[27] A. Ramkumar, P. J. Stappers, W. J. Niessen, S. Adebahr, T. Schimek-Jasch, U. Nestle, and Y. Song. "Using GOMS and NASA-TLX to Evaluate Human–Computer Interaction Process in Interactive Segmentation". In: *International Journal of Human–Computer Interaction* 33.2 (2017), pp. 123–134.

[28] V. Braun and V. Clarke. "Using thematic analysis in psychology". In: *Qualitative Research in Psychology* 3.2 (2006), pp. 77–101.

[29] A. Aselmaa, M. van Herk, A. Laprie, U. Nestle, I. Götz, N. Wiedenmann, T. Schimek-Jasch, F. Picaud, C. Syrykh, L. V. Cagetti, M. Jolnerovski, Y. Song, and R. H. Goossens. "Using a contextualized sensemaking model for interaction design: A case study of tumor contouring". In: *Journal of Biomedical Informatics* 65 (2017), pp. 145–158.

[30] C. L. Brouwer, R. J. H. M. Steenbakkers, J. Bourhis, W. Budach, C. Grau, V. Grégoire, M. van Herk, A. Lee, P. Maingon, C. Nutting, B. O'Sullivan, S. V. Porceddu, D. I. Rosenthal, N. M. Sijtsema, and J. A. Langendijk. "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.

[31] E. Aliotta, H. Nourzadeh, and J. Siebers. "Quantifying the dosimetric impact of organ-at-risk delineation variability in head and neck radiation therapy in the context of patient setup uncertainty". In: *Physics in Medicine & Biology* 64.13 (2019), p. 135020.

[32] J. Marewski and G. Gigerenzer. "Heuristic decision making in medicine". In: *Dialogues in clinical neuroscience* 14 (Mar. 2012), pp. 77–89. doi: 10.31887/DCNS.2012.14.1/jmarewski.

[33] M. Graber, R. Gordon, and N. Franklin. "Reducing Diagnostic Errors in Medicine: What's the Goal?" In: *Academic Medicine* 77.10 (2002).

[34] A. Tversky and D. Kahneman. "Judgment under Uncertainty: Heuristics and Biases". In: *Science* 185.4157 (1974), pp. 1124–1131.

[35] A. J. Chan, M. K. Islam, T. Rosewall, D. A. Jaffray, A. C. Easty, and J. A. Cafazzo. "Applying usability heuristics to radiotherapy systems". In: *Radiotherapy and Oncology* 102.1 (2012), pp. 142–147.

[36] B. S. Chera, M. Jackson, L. M. Mazur, R. Adams, S. Chang, K. Deschesne, T. Cullip, and L. B. Marks. "Improving Quality of Patient Care by Improving Daily Practice in Radiation Oncology". In: *Seminars in Radiation Oncology* 22.1 (2012), pp. 77–85.

[37] L. B. Marks, M. Jackson, L. Xie, S. X. Chang, K. D. Burkhardt, L. Mazur, E. L. Jones, P. Saponaro, D. LaChapelle, D. C. Baynes, and R. D. Adams. "The challenge of maximizing safety in radiation oncology". In: *Practical Radiation Oncology* 1.1 (2011), pp. 2–14.

[38] T. Heimann and H.-P. Meinzer. "Statistical shape models for 3D medical image segmentation: A review". In: *Medical Image Analysis* 13.4 (2009), pp. 543–563.

[39] M. Hermann and R. Klein. "A visual analytics perspective on shape analysis: State of the art and future prospects". In: *Computers & Graphics* 53 (2015), pp. 63–71.

[40] X. Gao, Y. Su, X. Li, and D. Tao. "A Review of Active Appearance Models". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 40.2 (2010), pp. 145–158. doi: 10.1109/TSMCC.2009.2035631.

[41] C. Chu, M. Oda, T. Kitasaka, K. Misawa, M. Fujiwara, Y. Hayashi, Y. Nimura, D. Rueckert, and K. Mori. "Multi-organ Segmentation Based on Spatially-Divided Probabilistic Atlas from 3D Abdominal CT Images". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*. Ed. by K. Mori, I. Sakuma, Y. Sato, C. Barillot, and N. Navab. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 165–172.

[42] T. LaBonte, C. Martinez, and S. A. Roberts. *We Know Where We Don't Know: 3D Bayesian CNNs for Credible Geometric Uncertainty*. 2020. arXiv: 1910.10793 [eess.IV]. url: https://arxiv.org/abs/1910.10793.

[43] P. Mody, N. Chaves-de-Plaza, K. Hildebrandt, R. van Egmond, H. de Ridder, and M. Staring. *Comparing Bayesian Models for Organ Contouring in Head and Neck Radiotherapy*. 2022. arXiv: 2111.01134 [eess.IV]. url: https://arxiv.org/abs/2111.01134.

[44] A. Top, G. Hamarneh, and R. Abugharbieh. "Active Learning for Interactive 3D Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011*. Ed. by G. Fichtinger, A. Martel, and T. Peters. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 603–610.

[45] M. B. Altman, J. A. Kavanaugh, H. O. Wooten, O. L. Green, T. A. DeWees, H. Gay, W. L. Thorstad, H. Li, and S. Mutic. “A framework for automated contour quality assurance in radiation therapy including adaptive techniques”. In: *Physics in Medicine & Biology* 60.13 (2015), p. 5199.

[46] H. Chen, S. Zhang, W. Chen, H. Mei, J. Zhang, A. Mercer, R. Liang, and H. Qu. “Uncertainty-Aware Multidimensional Ensemble Data Visualization and Exploration”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.9 (2015), pp. 1072–1086. doi: 10.1109/TVCG.2015.2410278.

[47] C. B. Hui, H. Nourzadeh, W. T. Watkins, D. M. Trifiletti, C. E. Alonso, S. W. Dutta, and J. V. Siebers. “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.

[48] J. Kalpathy-Cramer and C. Fuller. “Target Contour Testing/Instructional Computer Software (TaCTICS): A Novel Training and Evaluation Platform for Radiotherapy Target Delineation”. In: *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium* 2010 (Nov. 2010), pp. 361–5.

[49] C. McIntosh, I. Svistoun, and T. 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. doi: 10.1109/TMI.2013.2251421.

[50] D. J. Rhee, C. E. Cardenas, H. Elhalawani, R. McCarroll, L. Zhang, J. Yang, A. S. Garden, C. B. Peterson, B. M. Beadle, and L. E. Court. “Automatic detection of contouring errors using convolutional neural networks”. In: *Medical Physics* 46.11 (2019), pp. 5086–5097.

[51] V. Sandfort, K. Yan, P. M. Graffy, P. J. Pickhardt, and R. M. Summers. “Use of Variational Autoencoders with Unsupervised Learning to Detect Incorrect Organ Segmentations at CT”. In: *Radiology: Artificial Intelligence* 3.4 (2021), e200218.

[52] A. Al-Taie, H. K. Hahn, and L. Linsen. “Uncertainty estimation and visualization in probabilistic segmentation”. In: *Computers & Graphics* 39 (2014), pp. 48–59.

[53] J. Praßni, T. Ropinski, and K. Hinrichs. “Uncertainty-Aware Guided Volume Segmentation”. In: *IEEE Transactions on Visualization & Computer Graphics* 16.06 (2010), pp. 1358–1365. issn: 1941-0506. doi: 10.1109/TVCG.2010.208.

[54] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. “Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[55] C. Lundström, P. Ljung, A. Persson, and A. Ynnerman. “Uncertainty Visualization in Medical Volume Rendering Using Probabilistic Animation”. In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (2007), pp. 1648–1655. doi: 10.1109/TVCG.2007.70518.

[56] R. G. Raidou, F. J. J. Marcelis, M. Breeuwer, E. Gröller, A. Vilanova, and H. M. M. v. d. Wetering. “Visual Analytics for the Exploration and Assessment of Segmentation Errors”. In: *Eurographics Workshop on Visual Computing for Biology and Medicine*. Ed. by S. Bruckner, B. Preim, A. Vilanova, H. Hauser, A. Hennemuth, and A. Lundervold. The Eurographics Association, 2016. isbn: 978-3-03868-010-9. doi: 10.2312/vcbm.20161287.

[57] K. Furmanová, L. P. Muren, O. Casares-Magaz, V. Moiseenko, J. P. Einck, S. Pilskog, and R. G. Raidou. “PREVIS: Predictive visual analytics of anatomical variability for radiotherapy decision support”. In: *Computers & Graphics* 97 (2021), pp. 126–138.

[58] A. Saad, G. Hamarneh, and T. Möller. “Exploration and Visualization of Segmentation Uncertainty using Shape and Appearance Prior Information”. In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1366–1375. doi: 10.1109/TVCG.2010.152.

[59] O. Asan, A. E. Bayrak, and A. Choudhury. "Artificial Intelligence and Human Trust in Healthcare: Focus on Clinicians". In: *J Med Internet Res* 22.6 (2020), e15154.

[60] M. P. White, J. C. Cohrs, and A. S. Göritz. "Dynamics of Trust in Medical Decision Making: An Experimental Investigation into Underlying Processes". In: *Medical Decision Making* 31.5 (2011), pp. 710–720.

[61] M. Allnutt. "Human factors in accidents". In: *Quality and Safety in Health Care* 11.4 (2002), pp. 369–374.

[62] E. Persson, K. Barrafrem, A. Meunier, and G. Tinghög. "The effect of decision fatigue on surgeons' clinical decision making". In: *Health Economics* 28.10 (2019), pp. 1194–1203.

[63] Z. Kato. *Markov Random Fields in Image Segmentation*. Now Publishers, Inc., 2012. doi: 10.1561/2000000035.

[64] C. Rother, V. Kolmogorov, and A. Blake. "GrabCut: Interactive Foreground Extraction Using Iterated Graph Cuts". In: *ACM Trans. Graph.* 23 (Aug. 2004), pp. 309–314. doi: 10.1145/1186562.1015720.

[65] J. Dai, K. He, and J. Sun. "BoxSup: Exploiting Bounding Boxes to Supervise Convolutional Networks for Semantic Segmentation". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1635–1643. doi: 10.1109/ICCV.2015.191.

[66] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. "ScribbleSup: Scribble-Supervised Convolutional Networks for Semantic Segmentation". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2016, pp. 3159–3167. doi: 10.1109/CVPR.2016.344. url: https://doi.ieeecomputersociety.org/10.1109/CVPR.2016.344.

[67] K. Maninis, S. Caelles, J. Pont-Tuset, and L. V. Gool. "Deep Extreme Cut: From Extreme Points to Object Segmentation". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2018, pp. 616–625. doi: 10.1109/CVPR.2018.00071. url: https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00071.

[68] M. Amrehn, J. Glasbrenner, S. Steidl, and A. Maier. "Comparative Evaluation of Interactive Segmentation Approaches". In: *Bildverarbeitung für die Medizin 2016*. Ed. by T. Tolxdorff, T. M. Deserno, H. Handels, and H.-P. Meinzer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 68–73.

[69] R. Hebbalaguppe, K. McGuinness, J. Kuklyte, G. Healy, N. O'Connor, and A. Smeaton. "How interaction methods affect image segmentation: User experience in the task". In: *2013 1st IEEE Workshop on User-Centered Computer Vision (UCCV)*. 2013, pp. 19–24. doi: 10.1109/UCCV.2013.6530803.

[70] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann. "Visual Analysis of Spatial Variability and Global Correlations in Ensembles of Iso-Contours". In: *Computer Graphics Forum* 35.3 (2016), pp. 221–230.

[71] J. Bertholet, G. Anastasi, D. Noble, A. Bel, R. Leeuwen, T. Roggen, M. Duchateau, S. Pilskog, C. Garibaldi, N. Tilly, R. García-Mollá, J. Bonaque, U. Oelfke, M. Aznar, and B. Heijmen. "Patterns of practice for adaptive and real-time radiation therapy (POP-ART RT) part II: Offline and online plan adaption for interfractional changes". In: *Radiotherapy and Oncology* 153 (June 2020). doi: 10.1016/j.radonc.2020.06.017.

3

IMPLEMENTATION OF DELINEATION ERROR DETECTION SYSTEMS IN TIME-CRITICAL RADIOTHERAPY: DO AI-SUPPORTED OPTIMIZATION AND HUMAN PREFERENCES MEET?

Building upon the workflow insights of Chapter 2, this chapter explores how delineation error detection systems (DEDS) can address the navigation and segmentation error analysis steps in the quality assessment phase. By leveraging ensemble information and clinical features, DEDS aim to guide clinicians to slices containing potential segmentation anomalies, thereby reducing unnecessary navigation and editing. To evaluate DEDS' potential to streamline clinical workflows, we combined insights from user studies with two Dutch clinicians and a simulation model applied to retrospective data from 42 head and neck cancer patients. The results reveal key factors for successful

This chapter is based on the published paper: Chaves-de-Plaza, Nicolas F., P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, A. Vilanova, and R. van Egmond. "Implementation of delineation error detection systems in time-critical radiotherapy: Do AI-supported optimization and human preferences meet?" In: Cognition, Technology & Work (2024). doi: 10.1007/s10111-024-00784-4 *adoption, including the availability of clinically relevant prioritization metrics, the tolerability of trade-offs in segmentation accuracy, and the implementation of intuitive user interfaces. This chapter demonstrates how DEDS can reduce the quality assessment workload and lays the groundwork for integrating ensemble-based systems into clinical practice—a theme further explored in the ensemble-focused chapters that follow.*

3.1. INTRODUCTION

External beam radiotherapy (EBRT) is a widely used cancer treatment that relies on the precise delineation of tumors and organs-at-risk (OARs) to optimize radiation dose delivery. Manual delineation is laborious and time-consuming, hindering the adoption of time-sensitive therapies like adaptive proton therapy [1–3]. AI technologies such as deep learning-based auto-delineation can swiftly generate delineations from CT or MRI scans, reducing clinician workload and enhancing consistency [4–6]. However, AI-generated delineations often contain inaccuracies requiring quality assessment (QA) by clinicians [7].

As Fig. 3.1 illustrates, the QA process involves clinicians navigating auto-delineated image slices to identify and correct errors, a particularly demanding task for anatomically complex regions like the head and neck. Recently, delineation error detection systems (DEDS) have been proposed to streamline QA by highlighting areas likely to contain errors [8–10]. While these technologies promise to reduce QA time, their clinical implementation and impact on workflow efficiency remain underexplored.

This study aims to advance the clinical applicability of DEDS by addressing questions about the suitability of the DEDS workflow and

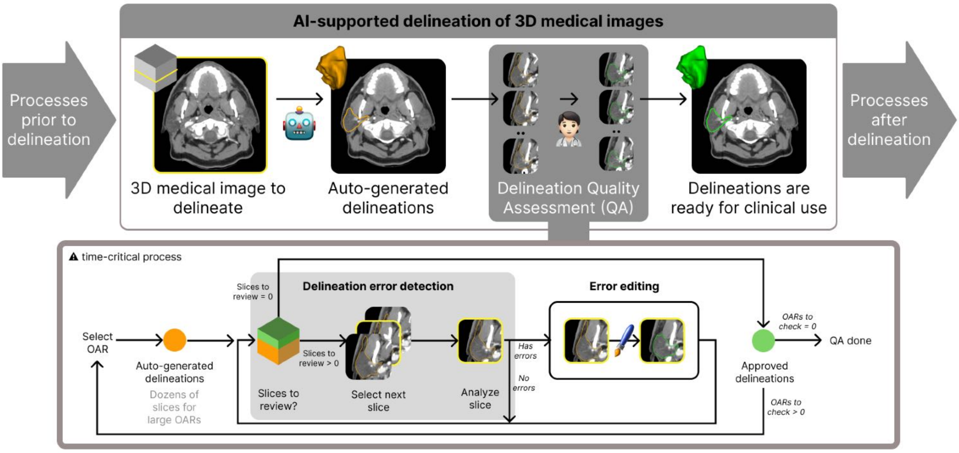

AI-supported delineation of 3D medical images
Processes prior to delineation
3D medical image to delineate
Auto-generated delineations
Delineation Quality Assessment (QA)
Delineations are ready for clinical use
Processes after delineation
time-critical process
Select OAR
Auto-generated delineations
Dozens of slices for large OARs
Slices to review = 0
Delineation error detection
Slices to review > 0
Slices to review?
Select next slice
Analyze slice
Has errors
No errors
Error editing
OARs to check = 0
QA done
Approved delineations
OARs to check > 0

Figure 3.1.: Overview of the AI-infused delineation workflow. The input is a set of 3D image volumes to delineate, a computerized tomography (CT) in the example. After generating the initial delineations with the AI, the clinician proceeds to perform a quality assessment (QA). The process has two tasks that alternate until there are no more errors: delineation error detection and editing.

its potential to expedite the QA process. We employed a mixed methods approach, starting with an observational user study involving a radiotherapy technologist and a radiation oncologist from Holland Proton Therapy Center (HollandPTC) to refine the DEDS workflow and validate several information sources for error detection and prioritization. This was followed by a simulation study that assessed the time-saving potential of various DEDS workflows across a diverse patient cohort with varying anatomies and error patterns.

The user study revealed a preference among the two clinicians for prioritizing errors based on clinical metrics, such as dose, over other forms of assistance with which they are less familiar. Further, DEDS assistance proved cumbersome, with the two clinicians expressing fatigue and confusion about the suggested slice orderings. These obstacles prompted the radiotherapy technologist to partially revert to a sequential slice-by-slice approach when navigating three-dimensional image volumes. Simulation results indicate that DEDS can improve the QA time-quality trade-off, although further refinement is needed for integration into clinical practice. This work sets a benchmark for DEDS evaluation and provides a simulation model that can be used to assess different error detection strategies.

3.2. RELATED WORK

Existing literature on user evaluation of radiotherapy software and workflows focuses on treatment planning process steps like delineation [11–13] and dose optimization [14, 15]. Particular to the case of delineation, research has focused on understanding the delineation workflow [16]; and investigating the effect of alternative image modalities [13] and delineation uncertainty [17], and usability of semi-automatic editing tools [18–20]. Recently introduced deep neural networks (DNNs) generating delineations of hundreds of OARs at once [4, 5] prompt clinics to create clinician-centric delineation quality assessment (QA) processes to identify and rectify DNNs inaccuracies [7].

This paper focuses on the delineation error detection QA subprocess. Delineation error detection systems (DEDS) can identify errors at various levels, from voxels to anatomical structures [21–25]. DEDS accelerate QA by directing attention to errors, reducing unnecessary scrutiny of clinically-acceptable delineations. For instance, some DEDS employ AIs to predict errors within slices based on auto-generated delineations and their uncertainty [8]. Recent developments even suggest a DEDS module that actively directs clinicians to the next slice for review based on predicted error extent [9] or predicted dosimetric impact [10]. Despite advances in DEDS, their clinical implementation and associated user experience challenges remain largely unaddressed issues.

In adaptive radiotherapy, clinicians prioritize areas based on dose distribution and patient malignancies [26]. Various studies explore the dosimetric impact of delineation errors [27–29]. Recent work introduces a DEDS that utilizes deformations of auto-generated delineations and dose prediction technologies to identify dosimetrically relevant areas for inspection [10]. We incorporate dose as a clinically relevant priority measure and discuss alternatives with the two clinicians in the study when dose information is unavailable.

3.3. MATERIALS AND METHODS

We used imaging data associated with a retrospective cohort of 42 head and neck cancer patients treated at Holland Proton Therapy Center (HollandPTC) between 2018 and 2020. The study from which the patient data was taken received IRB approval from Holland Proton Therapy Center (HollandPTC), and all patients provided informed consent. Data from three patients were employed for the user study and the complete cohort for the simulation study.

Fig. 3.2 presents an overview of the different types of three-dimensional images available per patient plus the additional ones we derived, like AI delineations and their uncertainty. In the remainder of this paper, we distinguish three-dimensional images, or image volumes, using a $\texttt{monospace}$ font. Unless stated otherwise, operations on pairs of volumes are applied voxel-wise, yielding a new volume (i.e., $\texttt{vol3} = \texttt{vol1} + \texttt{vol2}$). We use subscripts on the volume to index slices or voxels, which we specify in the text. For instance, $\texttt{vol}_s$ in the figure refers to the s^{th} 2D axial slice of $\texttt{vol}$.

3.3.1. IMAGING DATA

The top section of Fig. 3.2 displays slices of the patient's CT scan ($\texttt{image}$) and organ-at-risk (OAR) delineations ($\texttt{del}^*$) used for the original treatment planning. We define $\texttt{del}^*$ as delineation ground truth in our studies. In the user study, participants did not have access to $\texttt{del}^*$ while performing the error detection tasks. $\texttt{del}^*\texttt{(OAR)}$ represents the delineation of a specific OAR, which is a binary image with ones where the OAR lies and zeros otherwise. $\texttt{image}$ and $\texttt{del}^*$ have width, height and slice dimensions of sizes 512 × 512 × 195 voxels and spacing of 0.98 × 0.98 × 2 mm.

Each patient file also included the treatment $\texttt{dose}$ distribution volume, representing radiation deposition in space. In Fig. 3.2, brighter yellow and darker purple colors mean higher and lower dose values, respectively. We resampled the $\texttt{dose}$ to match the dimension sizes of $\texttt{image}$ and $\texttt{del}^*$. We include the $\texttt{dose}$ in our studies because the participants have an adaptive radiotherapy background, where the dose is used as a heuristic to determine which slices need more attention [10].

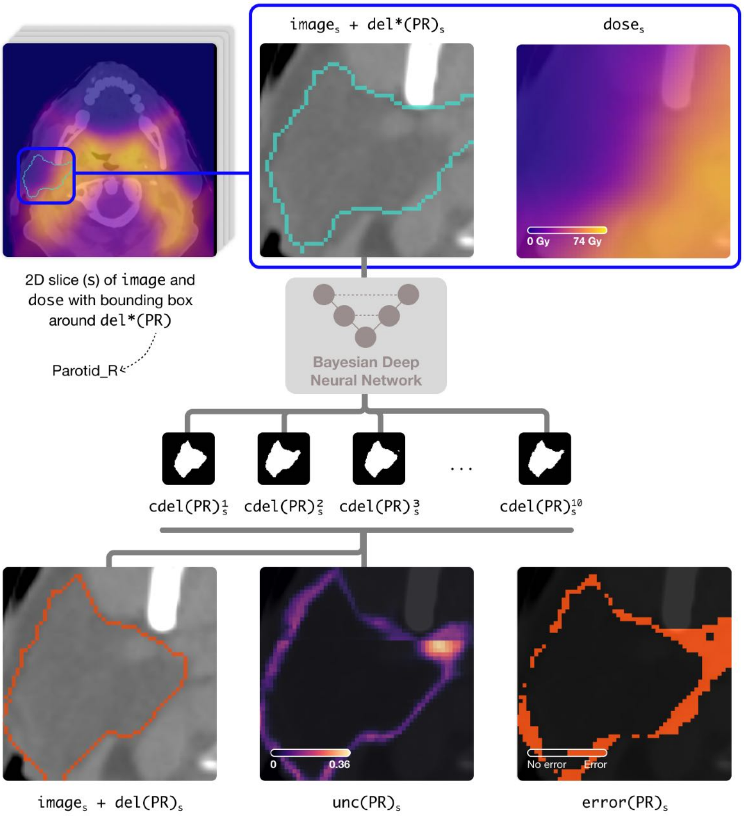


Figure 3.2.: Example of the information sources used in this paper for one of the patients in the HollandPTC dataset. The top row depicts a slice of the image and dose of the Parotid_R. We used a Bayesian Deep Neural Network to obtain ten delineation candidates based on the image. The bottom row depicts the information sources we derived based on these candidates.

In certain situations, metrics such as the distance to the target volumes may be more appropriate than the dose. Deciding to prioritize one over the other would necessitate rearranging the slices and consequently altering the workflow, which constitutes the primary focus of our paper.

For preprocessing, we cropped all three volumes using a bounding box centered at the brain stem with dimensions 240 × 240 × 80 voxel and spacing of 0.8 × 0.8 × 2.5 mm. Linear interpolation was applied to `image` and `dose`, while nearest-neighbor interpolation was used for `del`*. These preprocessing steps aligned the data with the input format expected by the AI.

3.3.2. AI DELINEATIONS, UNCERTAINTY AND ERROR

We fed the patient's $\texttt{image}$ in the HollandPTC dataset to a pre-trained state-of-the-art Bayesian deep neural network (the AI in this work), to generate ten candidate delineations for each input image. For this, we used the FlipOut model described in [30], which is based on the FocusNet architecture, employing a modified cross-entropy loss. The model generates delineation candidates by running ten times, each with a different set of weights sampled from a learned distribution. The network was trained on a subset of 33 patients of the MICCAI2015 head and neck dataset [31]. For each patient, there are delineations for nine OARs of which we used six: BrainStem, Mandible, parotid glands (Parotid_L and Parotid_R), and submandibular glands (Submand_L and Submand_R). We refer the reader to the original publication for more details about the network architecture and training.

Each AI-generated candidate $\texttt{cdel}^i$ with $i \in \{1, \ldots, 10\}$ is a label map volume, with each voxel having the ID of the OAR it belongs to (or zero if background). To aggregate the candidates into the predicted delineation $\texttt{del}$, we computed the voxel-wise median label:

$$\texttt{del} = \mathbb{M}(\texttt{cdel}^1, \ldots, \texttt{cdel}^{10}), \tag{3.1}$$

where $\mathbb{M}$ denotes the voxel-wise median function. $\texttt{del}$ is also a label map with the same dimensions and spacing as $\texttt{image}$. To obtain an OAR's predicted segmentation $\texttt{del(OAR)}$, it suffices to set voxels matching match a given OAR ID to one and the rest to zero. Note that the median operation can be thought of as performing a voxel-wise majority vote on the OAR IDs.

From the candidate delineations, we also calculated the AI's uncertainty $\texttt{unc}$ per OAR as the voxel-wise standard deviation of the OAR's candidates:

$$\texttt{unc(OAR)} = \sqrt{\frac{\sum_{i=1}^{10}(\texttt{cdel(OAR)}^{i} - \bar{\mu}\texttt{(OAR)})^2}{9}}, \tag{3.2}$$

where $\texttt{cdel(OAR)}^{i}$ represents the binary image of the OAR's i^{th} delineation candidate and $\bar{\mu}\texttt{(OAR)}$ the mean delineation for a specific OAR.

As the sample $\texttt{unc}$ slice in Fig. 3.2 illustrates, the computed uncertainty exhibits higher values (brighter spots) in image regions with challenging delineation, such as those lacking inter-tissue contrast. We prefer AI uncertainty over previous hand-engineered feature-based methods because it is readily available from the Bayesian network, requiring less domain-specific knowledge, and is correlated with delineation errors [8, 30]. Therefore, in our studies we adopt $\texttt{unc}$ as a proxy for delineation errors' location and extent.

The final information source we consider is the delineation error $\texttt{error}$, calculated as

$$\texttt{error}(\text{OAR}) = |\texttt{del}^*(\text{OAR}) - \texttt{del}(\text{OAR})|, \tag{3.3}$$

where $|\cdot|$ is the voxel-wise absolute value function. $\texttt{error}(\text{OAR})$ highlights areas where AI predictions and HollandPTC's delineations disagree. Note we do not differentiate between under and over-segmentation errors. Being an error proxy, $\texttt{unc}$ can suffer from false positives and negatives. In the studies, we use $\texttt{error}$ to provide an upper bound to the performance gains, assuming an optimal error detector. Finally, in the user study, we use $\texttt{error}$ as an additional information source to elicit discussion, allowing participants to contrast it with $\texttt{unc}$.

3.3.3. PER-SLICE SCORES

To enable priority sorting in the DEDS-assisted workflow, for an OAR we compute per-slice scores based on the $\texttt{unc}$, $\texttt{dose}$, and $\texttt{error}$. Computing the priority scores $\boldsymbol{p}(\text{OAR})$ of an OAR's slices entails applying an aggregation function to each slice of the OAR and collecting the values in an array:

$$\begin{aligned} \boldsymbol{p}(\text{OAR}) = \{&\text{agg}(\texttt{vol}(\text{OAR})_{s=1}), \\ &\text{agg}(\texttt{vol}(\text{OAR})_{s=2}), \\ &\ldots, \\ &\text{agg}(\texttt{vol}(\text{OAR})_{s=S})\}, \end{aligned} \tag{3.4}$$

where $\text{agg}(\cdot)$ takes as input a set of voxels (in this case those in an axial slice s) and outputs a number. For instance, to obtain the mean uncertainty score, we set $\texttt{vol}(\text{OAR}) = \texttt{unc}(\text{OAR})$ and $\text{agg} = \text{mean}$. We only consider voxels within $\texttt{del}^*(\text{OAR})$'s bounding box to avoid assigning scores to unrelated parts of the volume, like slices above and below the OAR. The assumption of correct bounding boxes before QA is not unreasonable, as inspecting and rectifying OARs' bounding boxes is an easy task that could be performed beforehand. In the user study, we considered the minimum (min), maximum (max), mean, and sum aggregation functions to enable discussion. In the simulation study, we focused on the most relevant ones from the user study.

3.4. USER STUDY: WORKFLOW COMPARISON

We conducted a two-part user study to investigate clinicians' current (part 1) and DEDS-assisted (part 2) workflows. In the following, we describe the study setup and then present and discuss the main findings, which inform the simulation study in the next section.

3.4.1. STUDY SETUP

PARTICIPANTS

A radiation oncologist (RO) and a radiotherapy technologist (RTT) from Holland Proton Therapy Center (HollandPTC), specialized in the head-and-neck area participated in our study. Both participants have several years of experience and perform delineation tasks routinely. TU Delft's IRB approved this research, and each clinician provided informed consent to be part of the study.

APPARATUS

The clinicians utilized the DEDS depicted in Fig. 3.3. We developed the custom DEDS software based on several sessions with two clinicians from Leiden University Medical Center and University Medical Center Utrecht. The design process is detailed in App. A. The DEDS incorporates functionality from standard delineation software like the list of OAR to review and a slice-based image viewer that allows inspecting the image volumes with interactions such as navigation, zooming, and panning. This functionality enables traditional error detection workflows. Additionally, as detailed next, the DEDS software implements functionality that permits clinicians to define and execute priority-based workflows.

A more detailed slice-level OAR explorer (slice explorer) allowed participants to inspect OARs' slices and sort them based on a priority score

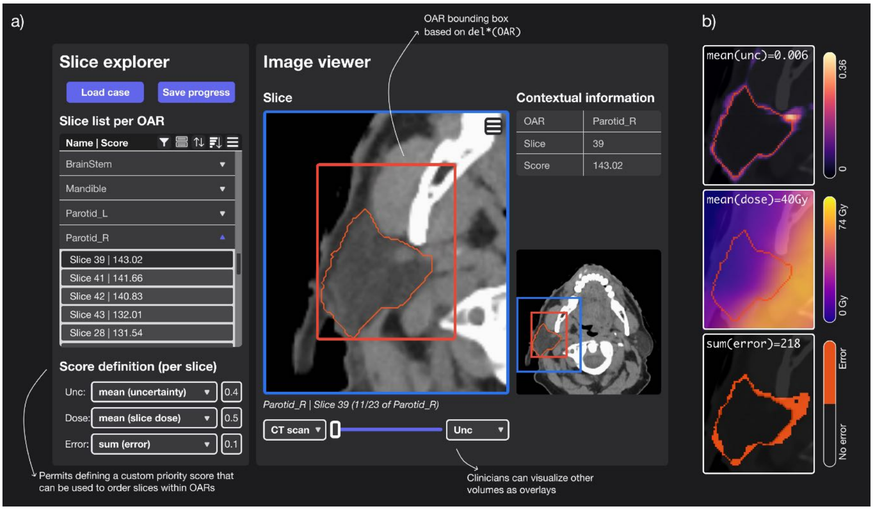
$$\begin{aligned}\boldsymbol{wp}\,(\texttt{OAR})_s = & w_1 \text{agg}_1\,(\texttt{unc}\,(\texttt{OAR})_s) + \\ & w_2 \text{agg}_2\,(\texttt{dose}\,(\texttt{OAR})_s) + \\ & w_3 \text{agg}_3\,(\texttt{error}\,(\texttt{OAR})_s),\end{aligned} \tag{3.5}$$

defined as weighted combination of `unc`, `dose`, and `error` scores. w_i represents weights, normalized to sum to one, and agg_i denotes aggregation functions. Participants selected their preferred aggregation functions and assigned them weights before starting part 2 of the study using the form in the score definition area of the DEDS' GUI in Fig. 3.3 (a). We allowed participants to define the priority score to elicit discussion about the relevance of different information sources and aggregation functions.

Although `unc` can be used as an error proxy, it is not the only option. For instance, the approach of [8] directly flags errors at the patch level. To facilitate richer discussions, we decided to permit participants to use the `error` and told them it was computed by an automatic method to prevent overreliance. Participants could overlay the volumes used for the score computation on the image viewer for closer inspection. A panel to the right of the image viewer (contextual information) provided details about the current slice, its score, and its location within the image. Fig. 3.3 (b) presents an example of the different information sources for slice $s = 11$ of OAR=Parotid_R.

PROCEDURE

The RTT and RO participated in a three-stage, 60-minute session. In the first stage, we presented the study's goal, introduced the clinicians to the DEDS, explained how to define priority scores based on weights and aggregation functions to sort OARs' slices, and let them interact with the DEDS to gain familiarity. In the second stage, the participants detected delineation errors without (part 1) and with (part 2) DEDS assistance. In part 1, participants performed their usual sequential error-finding workflow, permitting them to gain further familiarity with the tool before introducing assistance. For part 2, participants were instructed to use DEDS guidance by defining a priority score (as defined in Eq. 3.4) and using it to guide the order in which they visit OARs'



Slice 39 | 143.02
Slice 41 | 141.66
Slice 42 | 140.83
Slice 43 | 132.01
Slice 28 | 131.54

Score definition (per slice)

Unc: mean (uncertainty) 0.4
Dose: mean (slice dose) 0.5
Error: sum (error) 0.1

OAR	Parotid_R
Slice	39
Score	143.02

Parotid_R | Slice 39 (11/23 of Parotid_R)

Figure 3.3.: Custom DEDS software used in the study. (a) shows the graphical user interface. The main areas are the slice explorer and the image viewer. Using the score definition box, clinicians can define a slice ordering per OAR based on uncertainty, dose, and error information sources. (b) shows the available information sources for the currently displayed OAR (slice 11 of Parotid_R). It also presents the per-slice value obtained with the user-defined aggregation functions.

slices. In both parts, the participants were instructed to consider OARs' priorities when deciding which to address within a five-minute time window, chosen to induce the need to prioritize delineation errors. Furthermore, OARs were shown in the same order in the graphical user interface, and participants had to complete an OAR before moving on to the next. Finally, the participants were allowed to move back and forth between adjacent slices if needed for sense-making. Because rectifying errors is time-consuming and not within the scope of this study, we asked clinicians to instead indicate per slice if they would edit it via a keyboard shortcut. After finishing each task, we used a five-minute time slot to discuss the clinicians' experience using specific slices they marked as requiring editing, and, in part 1, to define the priority score. In the last 20-minute stage, we had a semi-structured discussion about participants' workflows, their choice of information sources for prioritization, and their experiences and challenges for DEDS adoption.

We used a subset (N=3) of HollandPTC's patients' data (D1, D2 and D3). D1 was used in the familiarization stage. The RO saw data from D2 and D3 in part 1 and part 2. The RTT observed D2 twice. This was unintentional and was not noticed until the data analysis phase. Therefore, we treated these sessions as independent observations, but we acknowledge this duplication as a limitation and have taken it into account when interpreting the results. Tab. 3.1 summarizes the structures considered in the user study analysis for D2 and D3. We do not include the mandible because clinicians tend to skip it due to its low clinical significance [32] and the clinicians' high confidence in AI auto-delineations for bony structures. Also note that the parotid glands demand the most effort, with their bounding boxes spanning more slices and containing more voxels per slice than the BrainStem and submandibular glands.

DATA ANALYSIS

We recorded the screen and the participant's spoken remarks in the sessions. From these, we transcribed clinicians' remarks and timestamped OAR changes, slice changes, and slices marked as "required editing". We recorded slice changes, yielding information about the order in which clinicians inspect the delineations in each condition. These interaction logs allowed us to reconstruct clinicians' workflows.

3.4.2. PART 1: NON-ASSISTED WORKFLOW

The RTT and RO conducted the error-finding task as in clinical practice. Fig. 3.4 shows the sequence of slices followed by the RTT and RO for the BrainStem (a) and Parotid_L (b). Fig. 3.4 (a.1) and (b.1) display the clinicians' and optimal slice change sequences using the per-slice

Table 3.1.: Overview of the organs-at-risk (OARs) considered for analysis. The table lists, for each OAR of each dataset, the number of slices and amount of voxels per slice its bounding box spans. It also lists the volume in mm^3 of the OAR's delineation ground truth `del*`. Bold entries indicate the OAR with the largest volume within each dataset.

	Dataset 2 (D2)			Dataset 3 (D3)		
OAR	Number of Slices	Voxels per Slice	Volume (mm^3)	Number of Slices	Voxels per Slice	Volume (mm^3)
BrainStem	25	1666	29963	25	1872	36037
Parotid_L	25	2688	35736	26	4104	36875
Parotid_R	**26**	**2912**	**36646**	**24**	**4292**	**39267**
Submand_L	18	1209	12498	16	1015	10410
Submand_R	17	1394	10970	17	928	9970

sum of errors as the priority score. The y-axis is trimmed to slices within the bounding box of `del*(OAR)` and sorts the slices based on their 3D position within the image volume. Despite opposing starting directions, both clinicians share similar navigation behavior, following a sequential approach (unlike the optimal sequence's "jumpy" behavior), with the RTT moving from bottom to top and the RO mostly in reverse. They frequently revisited adjacent slices to verify multi-slice errors, particularly in the slice range $[14, 19]$ of the BrainStem.

To compare the slice sequences of different workflows, we calculated the number of slice change interactions required to review slices suggested by a DEDS. A subset S of an OAR's slices consists of the $|S|$ slices exceeding the threshold. We evaluated the interactions needed for slice subsets of increasing size as the threshold decreased, including clinician workflows with redundant interactions removed and hypothetical scenarios: an optimal sequence ordered by decreasing erroneous voxels per slice, a worst-case sequence reversing the optimal, and five random permutations of the optimal sequence, with the mean and 95% confidence interval.

Fig. 3.4 (a.2) and (b.2) show slice change interactions as a function of suggested slice subset size for clinicians' workflows and hypothetical scenarios. The optimal workflow forms a diagonal line with a unit slope, indicating slice changes match the subset size. The worst-case scenario appears as a horizontal line since the highest error slice is reviewed last. Random samples lie between the optimal and worst-case scenarios, approaching the latter as the subset size grows, reflecting higher chances of critical slices appearing later. Clinicians' workflows generally deviate from the optimal path and often exceed the worst-case

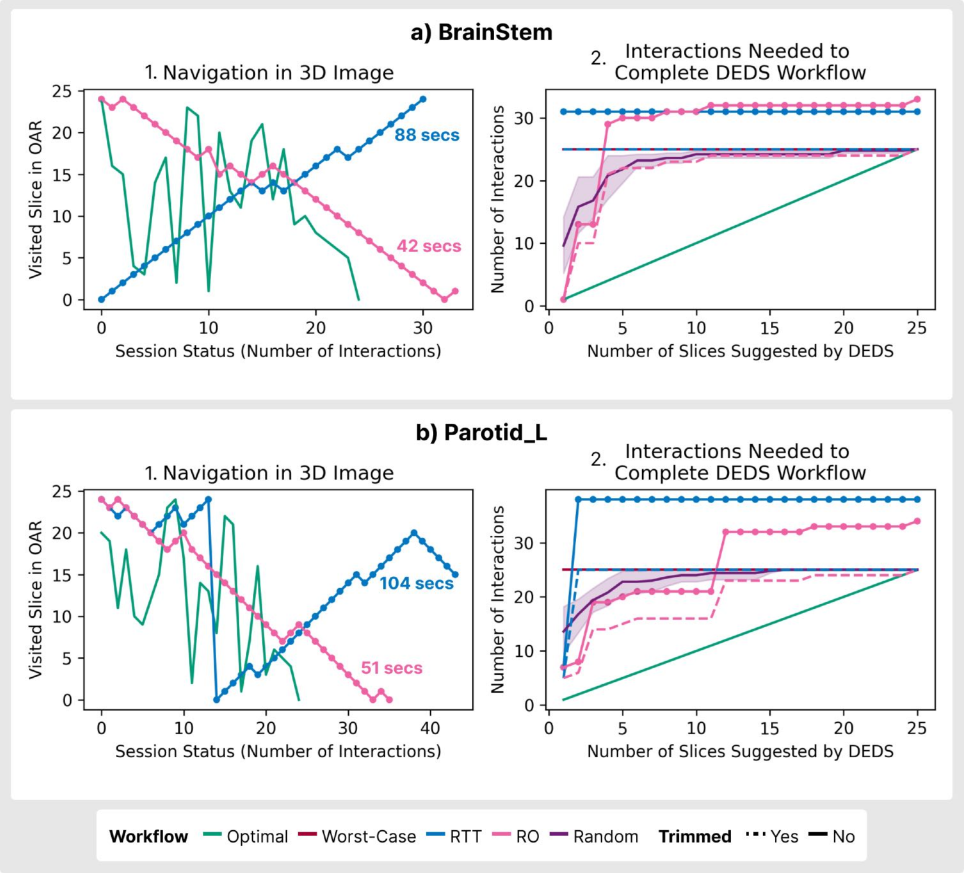


Figure 3.4.: Unassisted workflows for BrainStem (a) and Parotid_L (b) for the RTT and RO. (a.1) and (b.1) depict slice changes as the session progresses, and (a.2) and (b.2) show the interactions needed to complete a DEDS-suggested workflow, encompassing subsets of OAR's slices of increasing cardinality corresponding to decreasing threshold values for the prioritization scores. We compare the observed workflows with versions in which redundant interactions have been trimmed and with several hypothetical scenarios. The purple shaded area corresponds to the 95% confidence interval of the random scenario.

due to redundant interactions. Removing redundancy improves the RO's performance, aligning closer to or surpassing random workflows but still falling short of the optimal. The RTT's workflows remain near the worst-case, often missing critical slices early. The RO's workflows are faster than the RTT's, indicating shorter per-slice analysis times.

Table 3.2 compares the performance of different workflows for inspected OARs. Performance is quantified by the area under the curve relative to the optimal sequence, normalized per OAR. Scores closer to zero indicate near-optimal performance, while scores closer to one approach the worst-case scenario. Values above one reflect redundant interactions. Removing redundant visits (RTT' and RO') significantly improves scores. Trimmed RO workflows (RO') perform best, outperforming RTT and random sequences, but still deviate from the optimal, especially for the BrainStem and parotid glands, suggesting DEDS guidance could further reduce interactions and save time.

3.4.3. PART 2: DEDS-ASSISTED WORKFLOWS

In part 2, the RTT and RO were offered and instructed to use DEDS assistance to find slices that required attention. They started by defining a priority metric as a weighted combination of `unc`, `dose`, and `error` to sort the slices in priority order. Tab. 3.3 shows the combinations of information sources clinicians defined for different OARs. Both expressed reservations about the redundancy of uncertainty and error and their reliability in time-sensitive scenarios. This might be why clinicians emphasized dose-based risk measures, assigning lower weights to `unc` and `error`. Information sources, aggregation functions, and weights remained generally consistent across OARs. The sole exception was the aggregation function for dose-based slice scores for the parotid glands, where the RO adjusted it to the mean.

Table 3.2.: Performance of various error detection workflows. For a given workflow, its score corresponds to the difference between the areas under the workflow's and the optimal workflow's curves. The scores are normalized per OAR to provide comparable scores. The optimal and worst-case sequences have scores of zero and one, respectively. Clinicians' workflows with redundant slice visits removed are indicated by the apostrophe. Bold values highlight the smallest difference per OAR.

OAR	RTT	RTT'	RO	RO'	Random
BrainStem	1.50	1.00	1.32	**0.71**	0.81
Parotid_L	1.98	0.93	1.10	**0.52**	0.86
Parotid_R	-	-	1.11	**0.69**	0.84
Submand_L	-	-	0.30	**0.18**	0.80
Submand_R	-	-	0.21	**0.21**	0.75

Table 3.3.: Settings the RTT and RO used to define the priority score for sorting the slices of the different OARs in part 2 of the user study. agg denotes the aggregation functions and *w* the weights clinicians applied per information source and OAR.

	BrainStem				Parotid_L				Parotid_R			
	RTT		RO		RTT		RO		RTT		RO	
Source	agg	*w*	agg	*w*	agg	*w*	agg	*w*	agg	*w*	agg	*w*
`unc`	mean	0.50	mean	0.25	mean	0.50	mean	0.25	mean	0.50	mean	0.25
`dose`	max	0.50	max	0.65	max	0.50	max	0.65	max	0.50	mean	0.65
`error`	sum	0	sum	0.10	sum	0	sum	0.10	sum	0	sum	0.10

The RTT and RO found following the priority order to be cumbersome and fatiguing, echoing the RO's view that "jumping between slices is not logical" and disrupts the 3D perception. Fig. 3.5 illustrates this sentiment in the Parotid_R's workflow data. The RO (a) struggled with the initial sorting order provided by DEDS, leading to a reverse inspection (following ascending rather than descending priority score order), which led to a mirrored slice sequence as shown in (a.1). The RTT (b) intermittently followed the DEDS suggestions but often reverted to traditional navigation, as depicted in (b.1). Fig. 3.5 (a.2) and (b.2) show that deviations from the suggested sequence led to suboptimal performance. A similar pattern is evident in the BrainStem and parotid glands, as presented in Tab. 3.4. The trimmed RTT workflows (RTT') tend to perform better, as the RTT intermittently followed DEDS pointers, avoiding unnecessary slice visits, especially for the parotid glands.

Table 3.4.: Performance of various error detection workflows. For a given workflow, its score corresponds to the difference between the areas under the workflow's and the optimal workflow's curves. The scores are normalized per OAR to provide comparable scores. The optimal and worst-case sequences have scores of zero and one, respectively. Clinicians' workflows with redundant slice visits removed are indicated by the apostrophe. Bold values highlight the smallest difference per OAR.

OAR	RTT	RTT'	RO	RO'	Random
BrainStem	0.92	0.92	0.95	0.95	**0.42**
Parotid_L	0.57	**0.39**	1.08	1.00	0.40
Parotid_R	1.39	**0.34**	0.94	0.94	0.42

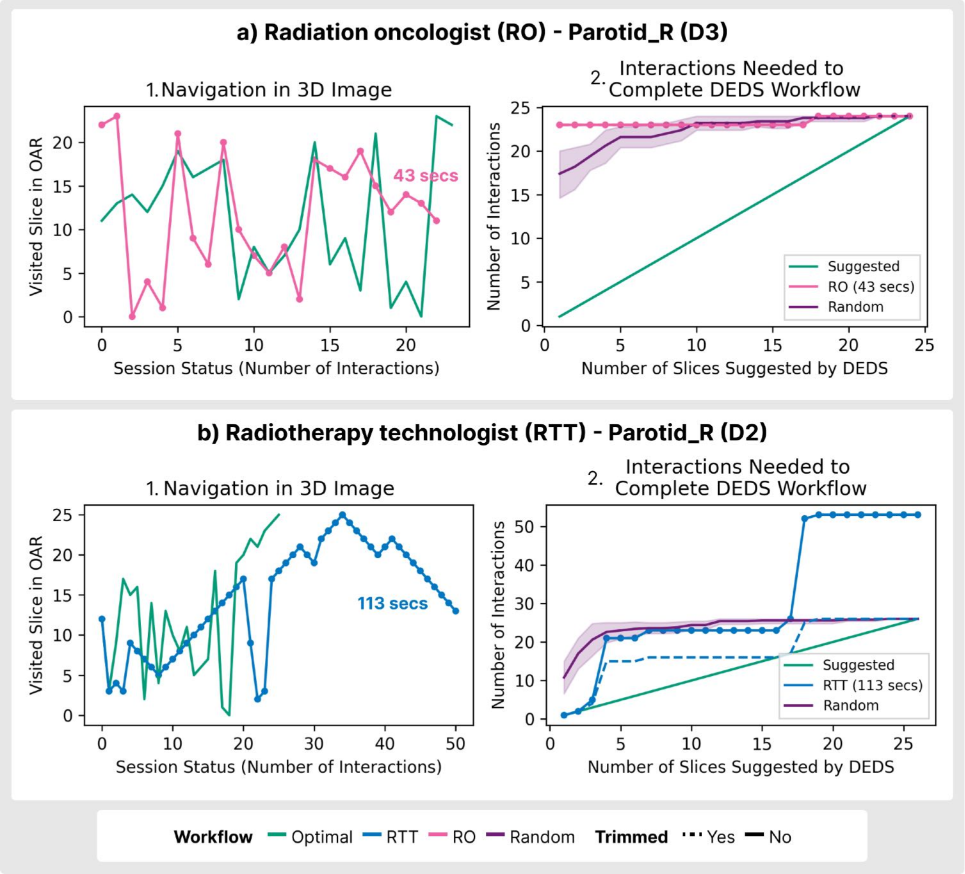


Figure 3.5.: Assisted workflows of the RO (a) and RTT (b) for Parotid_R. (a.1) and (b.1) depict slice changes as the session progresses, and (a.2) and (b.2) show the interactions needed to complete a DEDS-suggested workflow, encompassing subsets of OAR's slices of increasing cardinality corresponding to decreasing threshold values for the prioritization scores. We compare the observed workflows with versions in which redundant interactions have been trimmed and with several hypothetical scenarios. The purple shaded area corresponds to the 95% confidence interval of the random scenario.

3.4.4. DISCUSSION

Part 1 investigated clinicians' error detection workflows. Both the RO and RTT followed a sequential strategy, inspecting adjacent slices. They favored such workflow because it helps them to orientate spatially, leveraging their mental representations of the OARs. Nevertheless, the comparison of clinicians' workflows with other scenarios revealed that redundant and suboptimal sequences decrease their performance. Part 2 focused on investigating clinicians' use of DEDS systems. The RTT and RO had problems accepting this approach, complaining about fatigue, losing their spatial orientation, and, in the case of the RTT, repeatedly falling back to the sequential workflow. These issues need to be solved in the future since the workflow comparison again convincingly demonstrates that DEDS can reduce the number of needed interactions, which can also impact overall spent time.

Concerning the three information sources considered, both clinicians expressed their doubts regarding the intelligibility and trustworthiness of the uncertainty and error information sources. The dose was less problematic as an information source, likely due to participants' experience in adaptive radiotherapy where heuristics like stimating the delineation error's proximity to the tumor are employed. They mentioned that the maximum dose could provide a guiding signal because false positives and negatives are problematic in slices with a max dose higher than the OAR-specific limit. We leverage this observation in the next section to develop a computational model of the DEDS workflow.

The main limitation of the user study is the very small sample size. To test the insights from the user study on a larger dataset, we performed a quantitative evaluation of the DEDS-assisted QA workflow using a simulation approach. To this end, we introduce a computational model of the complete QA workflow, including analysis and editing, which we use to investigate the viability of DEDS workflows. Specifically, we analyze the impact of varying per-slice analysis times on overall QA performance for the complete HollandPTC dataset.

3.5. SIMULATION STUDY: ASSESSING DEDS-INDUCED TIME GAINS

3.5.1. SIMULATION SETUP

To examine the potential time savings achievable with DEDS, we compare DEDS workflows with the current unassisted clinical workflow. Fig. 3.6 depicts a computational model of the quality assessment process (QA). In our simulation, we consider three variations of this process that arise when using different slice sequences.

In the first variation (baseline), the simulated clinician begins either at the cranial or caudal slice with an equal probability ($Pr = 0.5$) and progresses towards the opposite end (next slice step), analyzing all slices. In the second (error) and third (dose) variations, the clinician visits the slices in order of their decreasing error extent and max dose,

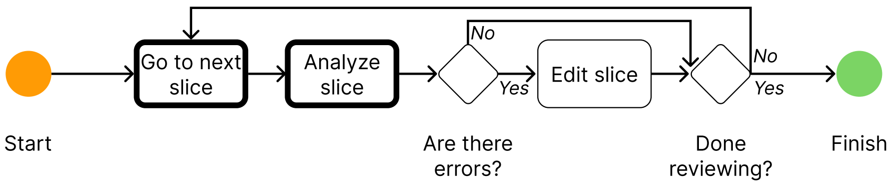


Figure 3.6.: Scheme of the delineation quality assessment (QA) process for an OAR. The analyze slice and edit slice rectangles have an associated time cost. The workflow variations we implement differ in the implementation of the go to next slice and analyze slice steps, which have a thicker border.

respectively. In these DEDS-assisted workflows, the simulated clinician evaluates a slice only if it has an error (error threshold equals zero) or its max dose exceeds a pre-set limit l(OAR), respectively. l(OAR) is an OAR-specific limit based on constraints proposed by [32]. In the error variations, we use delineation error instead of AI uncertainty because AI uncertainty serves as a proxy for delineation errors. By using the actual error, we simulate a best-case scenario where AI uncertainty perfectly identifies delineation errors.

For this study, the same OARs and bounding boxes per OAR as described in Sec. 3.3 were used. We preprocessed the `error` following the protocol proposed by [8] to remove tolerated errors. This filtering process excludes slices with errors that can be attributed to interobserver variation. An OAR's erroneous voxel is considered a tolerated error if it is within 2 pixels from the border of `del*(OAR)`, not part of a region of erroneous voxels of at least ten voxels in size, and not outside the top and bottom delineation limits. The slice metric we use for the error workflow is the sum of the non-tolerated erroneous voxels.

We use the dose as a proxy of the clinical significance of potential delineation errors for the patient's treatment. We selected the maximum as the aggregation function for the per-slice dose metric. [32] consider the mean dose, but we opted for the max based on the results of the user study. $\max(\texttt{dose}(\text{OAR})_s)$ is a more stringent constraint, representing a worst-case scenario for dosimetric deviations caused by erroneously delineated voxels in slice s. The max of the dose per slice indicates a lower risk in areas where the dose is consistently lower than the OAR's dosimetric constraint. The first three columns of Tab. 3.5 display the OARs, their max-dose constraints, and average slice numbers across patients for the baseline.

We simulate clinician behavior, relying on existing literature to estimate time costs for different steps. Based on [16], we model the time for analyzing a slice s in the baseline condition as $t_a(s) \sim \mathcal{N}(4.2, 3.2)$ seconds. For the error and dose conditions, we model the analysis time as $t_a^{\epsilon}(s) \sim \mathcal{N}(4.2 + \epsilon, 3.2)$ seconds. Here, ϵ represents the additional time required for analyzing DEDS suggestions, which are often not contiguous, resulting in jumps between non-sequential slices. In the simulation, we consider $\epsilon \in \{0, 4\}$ seconds, which allows us to assess the magnitude of the effect introduced by increasing analysis times. Finally, we assume a two-dimensional brush of size $bs = 10$ pixels for editing and model the time for editing a group of bs pixels as $t_{epix} \sim \mathcal{N}(1, 0.1)$ seconds. The time for editing a faulty slice is computed as $t_{ed}(s) = (t_{epix} \cdot \sum_{vox} \mathtt{errors}_s)/bs$. Note that the editing time modeling may vary depending on the editing tools used. In this case, we assume manual pixel brushing for simplicity. The total time per workflow execution is calculated as

$$T_{tot} = T_a + T_{ed} = \sum_{s \in S} t_a(s) + t_{ed}(s), \tag{3.6}$$

where S is the set of slices to review and T_a and T_{ed} represent the total analysis and editing time, respectively. To assess workflow quality, we calculated the percentage of attended errors for each workflow by dividing the sum of errors in the visited slices by the total amount of errors within the OAR's volume.

We conducted one hundred workflow runs for each combination of patient, OAR, and experimental condition (workflow variation) [1]. In the results, we aggregate numerical quantities like slice numbers and times across the workflow runs within each OAR of each patient to obtain a statistical overview of the differences between conditions.

3.5.2. RESULTS AND DISCUSSION

Tab. 3.5 aggregates slice numbers, percentages of attended errors, and total elapsed QA times across patients. The last row of the table indicates that, on average, the baseline workflow takes longer than dose-based workflows and the optimistic error-based one. In the baseline workflow, which takes 1034 seconds, the simulated clinician spends an average of 7.4 seconds per slice. In the error and dose workflows, the time per slice is 8.72 and 6.86 seconds for the optimistic scenario ($\epsilon = 0$) and 12.58 and 10.73 seconds for the pessimistic one ($\epsilon = 4$). Even if the time per slice is higher in the DEDS workflows, the total elapsed time generally turns out lower because clinicians do not need to check all slices. Regardless of the scenario, we observe a

[1] The simulation and analysis codes are available at `https://graphics.tudelft.nl/study-deds`

two-second difference in per-slice times between the dose-based and error-based workflows. These differences translate to total time savings of around two hundred seconds for both scenarios. However, these time gains come at the cost of quality. The table shows that while the baseline and error-based workflows addressed all errors, the dose-based ones only attended to 69% of them. A similar speed/quality tradeoff is expected if a higher threshold is used in the error-based workflows to limit the subset of slices for review. Focusing on individual OARs, we observe similar trends. Noteworthy are the BrainStem and the Mandible for which dose-based DEDS workflows obtain significant speedups. The dose-based workflows had the lowest percentage of addressed errors for the Mandible and BrainStem, indicating that many slices were skipped because they did not exceed the dosimetric constraints. This prioritization strategy, along with the larger size of these structures, accounts for the observed time savings. Skipping more slices, especially those with significant errors, reduces analysis and editing times but compromises delineation quality [26].

Focusing on the difference between scenarios, it is possible to observe how increasing the difficulty of the slice analysis task, and consequently, the time it takes leads to longer T_{tot}. Although the pessimistic dose scenario is competitive with the baseline, the error one significantly exceeds it. At the OAR level, we note that larger structures like the BrainStem and the Mandible, although closer to the baseline, still outperform it in most cases. This shows that, even with increased analysis times, DEDS can be particularly time-saving when used to review large anatomical structures, at the expense of confusing clinicians as seen in the user study.

To understand the contributions of the analysis (T_a) and editing (T_{ed}) times to the total QA time, in Fig 3.7 we visualize the total analysis (a) and editing (b) times per OAR per patient averaged across simulation runs. Each column of gray horizontal lines within an OAR's area corresponds to a simulated condition, denoted by the color of the diamond on the column. Each line corresponds to the average time per patient and the diamond presents the average across patients. In general, we observe that in the optimistic scenarios, the analysis times are consistently below the baseline. In the pessimistic scenario, DEDS analysis times are less favorable but stay close to the baseline for larger structures like the BrainStem and the Mandible, a similar trend to the one we observed for T_{tot} before. Except for the BrainStem, the dose-driven workflow consistently requires more time than the error-driven one for $\epsilon = 0$ and $\epsilon = 4$. This indicates that the max($\texttt{dose}$) criteria designate more slices as high-risk compared to error-free slices.

Concerning editing times, the figure indicates that the baseline and the error-based DEDS workflows perform similarly because, without a priority metric or error tolerance, the simulated clinician has to amend all the delineation errors in the error-based workflows. In contrast, the dose-based DEDS workflows are faster because they focus solely on slices with a high max dose, which are not necessarily the ones with the errors that take the longest to edit. In line with the results in Tab. 3.5, the improved performance of dose-based workflows is notable for the BrainStem and the Mandible, which are the largest structures and, therefore, tend to have more extensive erroneous regions. Finally, note that the times between scenarios do not change because we assumed the editing mechanism remains the same and is unaffected by the slice sequence.

In summary, the results of the simulation study suggest that DEDS workflows can reduce QA times. As the results for the dose-based workflows show, more significant time gains can be achieved by using more stringent thresholds to select the subset of slices to review at the cost of decreased delineation quality. This reduction in quality might be acceptable if it can be established that the bypassed errors are not clinically relevant. Our findings show diminishing DEDS advantages over the baseline workflow for smaller structures and when $\epsilon > 0$. Therefore, it is essential to reduce analysis time to justify the practical use of DEDS.

3.6. DISCUSSION

In this paper, we evaluated the clinical suitability of delineation error detection systems (DEDS). In particular, can DEDS speed up the Quality Assessment process without losing quality? To this end, we co-designed a DEDS with two experienced head and neck radiation oncologists from Utrecht University Medical Center and Leiden University Medical Center. The system was then used by two clinicians from HollandPTC to perform the assisted and unassisted DEDS workflows based on slice-wise statistics of the uncertainty, dose, and error. Based on insights from the user study, we addressed the question of whether DEDS can contribute to speeding up the clinical QA workflow using a simulation approach. A contribution of this work is a computational model of the QA process, which we used to simulate and compare several workflows. Researchers can use and extend this model to benchmark novel and existing DEDS proposals.

In the user study, we identified two key challenges to DEDS adoption. First, the information sources require refinement. Clinicians appreciated using dose information for its clarity, as it helped filter out clinically insignificant slices, but found the uncertainty and error metrics confusing, unnecessary, and potentially unreliable. This issue might be addressed by allowing more time for familiarization, introducing clearer indicators of uncertainty, and enhancing system-user compatibility in clinical settings [33–35]. Second, DEDS workflows often require navigating between non-contiguous slices, which clinicians found cumbersome and fatiguing. This navigation mode led clinicians to revert to conventional, sequential slice inspection, increasing the number of interactions. The challenge of maintaining a mental frame when jumping between slices could explain this behavior [16]. Providing less intrusive guidance or better tools to update clinicians' mental models could alleviate these issues [36].

The simulation study showed that DEDS can improve QA times over the current baseline, especially for large anatomical structures where only a subset of slices is relevant according to a predefined metric. Nevertheless, considering smaller subsets of potentially non-adjacent slices poses two challenges. First, analysis times increase because clinicians cannot inspect slices sequentially. A mitigation strategy could be to offer clinicians chunks of contiguous slices to allow more effective sense-making. Second, and perhaps more critical for the adoption of DEDS-based workflows, it should be possible to be certain that bypassed errors are not clinically relevant—a non-trivial challenge that requires improving AI uncertainty estimates and developing clinically relevant metrics [10]. For instance, DEDS could leverage clinical measurements or heuristics like distance to target volumes as a priority metric when the error or dose are unavailable. The proposed framework can directly accommodate new metrics by defining a per-slice aggregation and a weight, allowing for combination with other metrics if needed.

Finally, there are several future work avenues. First, the present study applies to OARs, but other high-priority structures like target volumes and elective lymph nodes could also be considered. Target volumes likely face challenges to adoption because clinicians are less willing to forego reviewing all slices due to the high risk they represent to the patient. For example, missing errors in target volumes could directly impact treatment outcomes, making clinicians cautious about skipping slices. Lymph node fields are more promising because of their large extent (which makes them cumbersome to delineate), high priority, and relative stability across the population, facilitating the recent development of auto-delineation technologies [37]. Second, the user and simulation studies could be extended to include other auto-delineation AIs and anatomical regions, which might have different error modes. Finally, the computational model of the QA process can be enriched, such as by using skewed distributions for modeling reaction times, which can be more appropriate but need substantial empirical data to estimate their parameters [38].

3.7. CONCLUSION

This study evaluated delineation error detection systems (DEDS) for improving the Quality Assessment (QA) process in clinical settings. A user study identified two main challenges that must be addressed to increase DEDS' adoption. First, clinicians preferred dose-based prioritization for error detection, finding it more intuitive than other metrics like uncertainty and error, which were seen as confusing and less reliable. Second, the non-sequential navigation required by DEDS disrupted clinicians' natural workflow, making it harder to make sense of the DEDS' suggestions. A computational model was introduced to benchmark different DEDS workflows. Simulations showed that DEDS could significantly reduce QA times, particularly for large structures, but this speed-up comes at the cost of delineation quality. Therefore, improving the accuracy of error proxies, such as AI uncertainty estimates, and developing metrics to assess the clinical significance of errors are crucial. Researchers can use and extend the computational model to further evaluate and refine DEDS.

Table 3.5.: Results of the simulation study conducted on a retrospective cohort of N=42 patients. The table lists the organs-at-risk (OARs) considered in the study and their dosimetric limits in Grays (Gy). For each workflow variation, it provides the average and standard deviation of the number of slices reviewed by the simulated clinicians and the percentage of errors addressed. For the total time taken to complete the QA process, results are further detailed by scenario within each workflow. Decimal places are omitted for clarity.

	l_{OAR}	Number of Slices			Attended Errors (%)			Total Elapsed Time (Seconds)				
OAR	(Gy)	*baseline*	*error*	*dose*	*baseline*	*error*	*dose*	*baseline*	*error*($\epsilon = 0$)	*error*($\epsilon = 4$)	*dose*($\epsilon = 0$)	*dose*($\epsilon = 4$)
BrainStem	54	23±3	18±5	7±6	100±0	100±0	20±21	182±94	157±100	225±111	48±47	76±70
Mandible	72	35±3	18±10	18±11	100±0	100±0	42±38	277±405	203±421	270±437	111±86	180±124
Parotid_L	26	27±3	21±5	22±8	100±0	100±0	79±30	175±44	150±53	231±69	144±63	231±90
Parotid_R	26	26±3	22±5	22±8	100±0	100±0	83±29	190±55	169±62	253±77	160±71	244±96
Submand_L	35	14±2	10±3	13±4	100±0	100±0	93±26	111±39	93±47	134±57	102±48	153±59
Submand_R	35	14±2	10±3	13±4	100±0	100±0	92±25	101±30	82±34	121±44	94±39	146±53
Total	-	140±9	98±19	96±30	100±0	100±0	68±19	1036±445	855±474	1233±516	659±227	1031±333

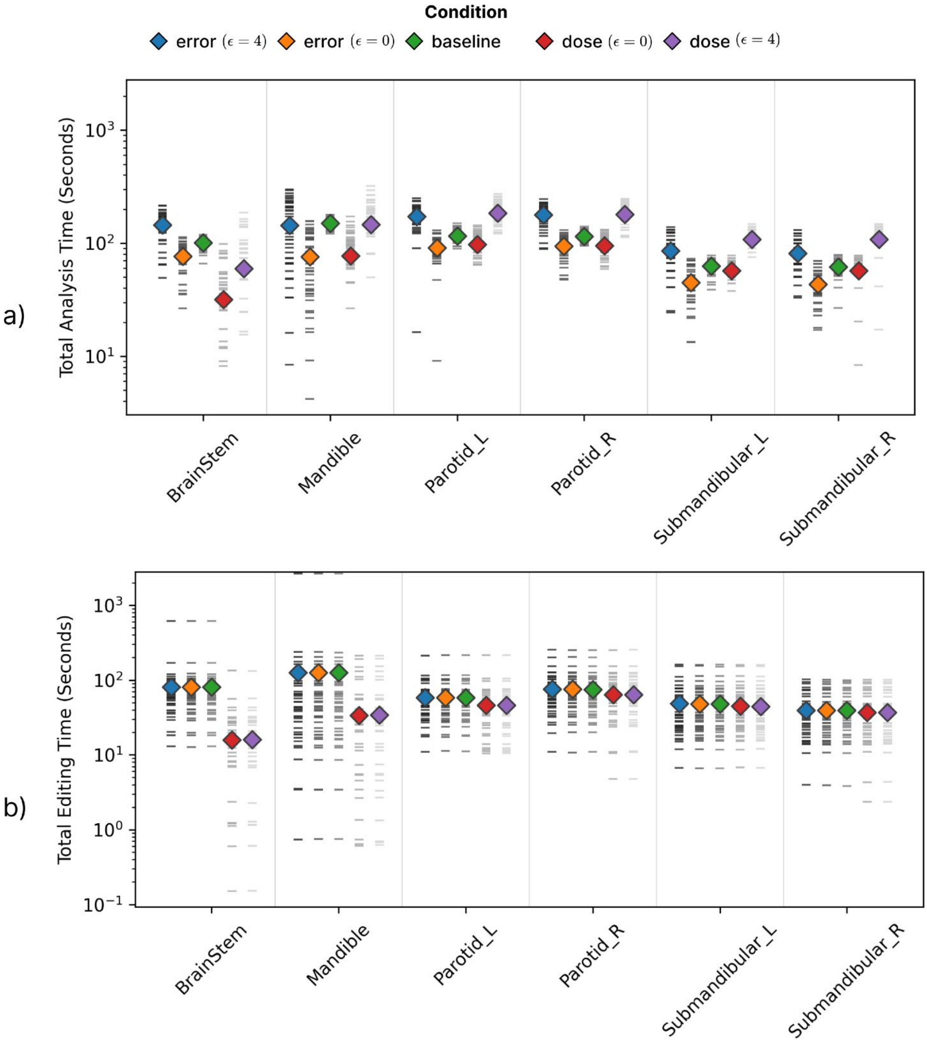


Figure 3.7.: Mean total analysis (a) and editing (b) times per OAR per patient in the cohort for the five simulated conditions. Each column within an OAR's area corresponds to the condition indicated by the color of the diamond. Gray horizontal lines within each column correspond to the patient's times, averaged across simulation runs. The colored diamond indicates the mean time per condition. The y-axis uses a logarithmic scale to enhance comparability and reduce empty space in the plot. Note that the y-axes of the two subplots have different ranges.

REFERENCES

[1] F. Albertini, M. Matter, L. Nenoff, Y. Zhang, and A. Lomax. “Online daily adaptive proton therapy”. In: *The British Journal of Radiology* 93.1107 (2020), p. 20190594.

[2] J.-J. Sonke, M. Aznar, and C. Rasch. “Adaptive Radiotherapy for Anatomical Changes”. In: *Seminars in Radiation Oncology* 29.3 (2019). Adaptive Radiotherapy and Automation, pp. 245–257. issn: 1053-4296. doi: https://doi.org/10.1016/j.semradonc.2019.02.007. url: https://www.sciencedirect.com/science/article/pii/S1053429619300165.

[3] P. Castadot, J. A. Lee, X. Geets, and V. Grégoire. “Adaptive Radiotherapy of Head and Neck Cancer”. In: *Seminars in Radiation Oncology* 20.2 (2010). Adaptive Radiotherapy, pp. 84–93. issn: 1053-4296. doi: https://doi.org/10.1016/j.semradonc.2009.11.002. url: https://www.sciencedirect.com/science/article/pii/S1053429609000769.

[4] S. Nikolov, S. Blackwell, A. Zverovitch, R. Mendes, M. Livne, J. De Fauw, Y. Patel, C. Meyer, H. Askham, B. Romera-Paredes, C. Kelly, A. Karthikesalingam, C. Chu, D. Carnell, C. Boon, D. D’Souza, S. A. Moinuddin, B. Garie, Y. McQuinlan, S. Ireland, K. Hampton, K. Fuller, H. Montgomery, G. Rees, M. Suleyman, T. Back, C. O. Hughes, J. R. Ledsam, and O. Ronneberger. “Clinically Applicable Segmentation of Head and Neck Anatomy for Radiotherapy: Deep Learning Algorithm Development and Validation Study”. In: *J Med Internet Res* 23.7 (July 2021), e26151. issn: 1438-8871. doi: 10.2196/26151. url: http://www.ncbi.nlm.nih.gov/pubmed/34255661.

[5] C. E. Cardenas, J. Yang, B. M. Anderson, L. E. Court, and K. B. Brock. “Advances in Auto-Segmentation”. In: *Seminars in Radiation Oncology* 29.3 (2019). Adaptive Radiotherapy and Automation, pp. 185–197. issn: 1053-4296. doi: https://doi.org/10.1016/j.semradonc.2019.02.001. url: https://www.sciencedirect.com/science/article/pii/S1053429619300104.

[6] J.-J. Sonke, M. Aznar, and C. Rasch. “Adaptive Radiotherapy for Anatomical Changes”. In: *Seminars in Radiation Oncology* 29.3 (2019). Adaptive Radiotherapy and Automation, pp. 245–257. issn: 1053-4296. doi: https://doi.org/10.1016/j.semradonc.2019.02.007. url: https://www.sciencedirect.com/science/article/pii/S1053429619300165.

[7] L. Vandewinckele, M. Claessens, A. Dinkla, C. Brouwer, W. Crijns, D. Verellen, and W. van Elmpt. “Overview of artificial intelligence-based applications in radiotherapy: Recommendations for implementation and quality assurance”. In: *Radiotherapy and Oncology* 153 (2020). Physics Special Issue: ESTRO Physics Research Workshops on Science in Development, pp. 55–66. issn: 0167-8140. doi: https://doi.org/10.1016/j.radonc.2020.09.008. url: https://www.sciencedirect.com/science/article/pii/S0167814020307805.

[8] J. Sander, B. D. de Vos, and I. Išgum. “Automatic segmentation with detection of local segmentation failures in cardiac MRI”. In: *Scientific Reports* 10.1 (Dec. 10, 2020), p. 21769. doi: 10.1038/s41598-020-77733-4. url: https://doi.org/10.1038/s41598-020-77733-4.

[9] T. Zhou, L. Li, G. Bredell, J. Li, J. Unkelbach, and E. Konukoglu. “Volumetric memory network for interactive medical image segmentation”. In: *Medical Image Analysis* 83 (2023), p. 102599. issn: 1361-8415. doi: https://doi.org/10.1016/j.media.2022.102599. url: https://www.sciencedirect.com/science/article/pii/S1361841522002316.

[10] B. Roberfroid, J. A. Lee, X. Geets, E. Sterpin, and A. M. Barragán-Montero. “DIVE-ART: A tool to guide clinicians towards dosimetrically informed volume editions of automatically segmented volumes in adaptive radiation therapy”. In: *Radiotherapy and Oncology* 192 (2024), p. 110108.

[11] J. Kalpathy-Cramer, M. Awan, S. Bedrick, C. R. N. Rasch, D. I. Rosenthal, and C. D. Fuller. “Development of a Software for Quantitative Evaluation Radiotherapy Target and Organ-at-Risk Segmentation Comparison”. In: *Journal of Digital Imaging* 27.1 (2014), pp. 108–119.

[12] R. J. Steenbakkers, J. C. Duppen, I. Fitton, K. E. Deurloo, L. Zijp, A. L. Uitterhoeve, P. T. Rodrigus, G. W. Kramer, J. Bussink, K. D. Jaeger, J. S. Belderbos, A. A. Hart, P. J. Nowak, M. van Herk, and C. R. Rasch. “Observer variation in target volume delineation of lung cancer related to radiation oncologist–computer interaction: A ‘Big Brother’ evaluation”. In: *Radiotherapy and Oncology* 77.2 (2005), pp. 182–190.

[13] R. J. Steenbakkers, J. C. Duppen, I. Fitton, K. E. Deurloo, L. J. Zijp, E. F. Comans, A. L. Uitterhoeve, P. T. Rodrigus, G. W. Kramer, J. Bussink, K. De Jaeger, J. S. Belderbos, P. J. Nowak, M. van Herk, and C. R. Rasch. “Reduction of observer variation using matched CT-PET for lung cancer delineation: A three-dimensional analysis”. In: *International Journal of Radiation Oncology*Biology*Physics* 64.2 (2006), pp. 435–448.

[14] L. M. Mazur, P. R. Mosaly, L. M. Hoyle, E. L. Jones, B. S. Chera, and L. B. Marks. “Relating physician’s workload with errors during radiation therapy planning”. In: *Practical Radiation Oncology* 4.2 (2014), pp. 71–75.

[15] L. M. Mazur, P. R. Mosaly, L. M. Hoyle, E. L. Jones, and L. B. Marks. “Subjective and objective quantification of physician’s workload and performance during radiation therapy planning tasks”. In: *Practical Radiation Oncology* 3.4 (2013), e171–e177.

[16] A. Aselmaa, M. van Herk, A. Laprie, U. Nestle, I. Götz, N. Wiedenmann, T. Schimek-Jasch, F. Picaud, C. Syrykh, L. V. Cagetti, M. Jolnerovski, Y. Song, and R. H. Goossens. “Using a contextualized sensemaking model for interaction design: A case study of tumor contouring”. In: *Journal of Biomedical Informatics* 65 (2017), pp. 145–158.

[17] F. C. Maruccio, W. Eppinga, M.-H. Laves, R. F. Navarro, M. Salvi, F. Molinari, and P. Papaconstadopoulos. “Clinical assessment of deep learning-based uncertainty maps in lung cancer segmentation”. In: *Physics in Medicine & Biology* 69.3 (Jan. 2024), p. 035007.

[18] A. Aselmaa, M. van Herk, Y. Song, R. H. M. Goossens, and A. Laprie. “The influence of automation on tumor contouring”. In: *Cognition, Technology & Work* 19.4 (2017), pp. 795–808.

[19] A. Ramkumar, J. Dolz, H. A. Kirisli, S. Adebahr, T. Schimek-Jasch, U. Nestle, L. Massoptier, E. Varga, P. J. Stappers, W. J. Niessen, and Y. Song. “User Interaction in Semi-Automatic Segmentation of Organs at Risk: a Case Study in Radiotherapy”. In: *Journal of Digital Imaging* 29.2 (2016), pp. 264–277.

[20] A. Ramkumar, P. J. Stappers, W. J. Niessen, S. Adebahr, T. Schimek-Jasch, U. Nestle, and Y. Song. “Using GOMS and NASA-TLX to Evaluate Human–Computer Interaction Process in Interactive Segmentation”. In: *International Journal of Human–Computer Interaction* 33.2 (2017), pp. 123–134.

[21] M. B. Altman, J. A. Kavanaugh, H. O. Wooten, O. L. Green, T. A. DeWees, H. Gay, W. L. Thorstad, H. Li, and S. Mutic. “A framework for automated contour quality assurance in radiation therapy including adaptive techniques”. In: *Physics in Medicine & Biology* 60.13 (June 2015), p. 5199. doi: 10.1088/0031-9155/60/13/5199. url: https://dx.doi.org/10.1088/0031-9155/60/13/5199.

[22] C. B. Hui, H. Nourzadeh, W. T. Watkins, D. M. Trifiletti, C. E. Alonso, S. W. Dutta, and J. V. Siebers. “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. doi: https://doi.org/10.1002/mp.12835. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.12835. url: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12835.

[23] D. J. Rhee, C. E. Cardenas, H. Elhalawani, R. McCarroll, L. Zhang, J. Yang, A. S. Garden, C. B. Peterson, B. M. Beadle, and L. E. Court. “Automatic detection of contouring errors using convolutional neural networks”. In: *Medical Physics* 46.11 (2019), pp. 5086–5097. doi: https://doi.org/10.1002/mp.13814. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.13814. url: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.13814.

[24] V. Sandfort, K. Yan, P. M. Graffy, P. J. Pickhardt, and R. M. Summers. “Use of Variational Autoencoders with Unsupervised Learning to Detect Incorrect Organ Segmentations at CT”. In: *Radiology: Artificial Intelligence* 3.4 (2021). PMID: 34350410, e200218. doi: 10.1148/ryai.2021200218. eprint: https://doi.org/10.1148/ryai.2021200218. url: https://doi.org/10.1148/ryai.2021200218.

[25] P. Mody, N. F. Chaves-de-Plaza, K. Hildebrandt, and M. Staring. “Improving Error Detection in Deep Learning Based Radiotherapy Autocontouring Using Bayesian Uncertainty”. In: *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging: 4th International Workshop, UNSURE 2022, Held in Conjunction with MICCAI 2022, Singapore, September 18, 2022, Proceedings*. Singapore, Singapore: Springer-Verlag, 2022, pp. 70–79. isbn: 978-3-031-16748-5. doi: 10.1007/978-3-031-16749-2_7. url: https://doi.org/10.1007/978-3-031-16749-2_7.

[26] N. F. Chaves-de-Plaza, P. P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, and van René van Egmond. “Towards fast human-centred contouring workflows for adaptive external beam radiotherapy”. In: *Proceedings of the Human Factors and Ergonomics Society Europe Chapter 2022 Annual Conference*. 2022. url: https://www.hfes-europe.org/enhancing-safety-critical-performance/.

[27] H. Guo, J. Wang, X. Xia, Y. Zhong, J. Peng, Z. Zhang, and W. Hu. “The dosimetric impact of deep learning-based auto-segmentation of organs at risk on nasopharyngeal and rectal cancer”. In: *Radiation Oncology* 16.1 (2021), p. 113.

[28] L. Mövik, A. Bäck, and N. Pettersson. “Impact of delineation errors on the estimated organ at risk dose and of dose errors on the normal tissue complication probability model”. In: *Medical Physics* 50.3 (2023), pp. 1879–1892.

[29] W. van Rooij, M. Dahele, H. Ribeiro Brandao, A. R. Delaney, B. J. Slotman, and W. F. Verbakel. “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.

[30] P. P. Mody, N. Chaves-de-Plaza, K. Hildebrandt, R. van Egmond, H. de Ridder, and M. Staring. “Comparing Bayesian models for organ contouring in head and neck radiotherapy”. In: *Medical Imaging 2022: Image Processing*. Ed. by O. Colliot and I. Išgum. Vol. 12032. International Society for Optics and Photonics. SPIE, 2022, 120320F. doi: 10.1117/12.2611083. url: https://doi.org/10.1117/12.2611083.

[31] P. F. Raudaschl, P. Zaffino, G. C. Sharp, M. F. Spadea, A. Chen, B. M. Dawant, T. Albrecht, T. Gass, C. Langguth, M. Lüthi, F. Jung, O. Knapp, S. Wesarg, R. Mannion-Haworth, M. Bowes, A. Ashman, G. Guillard, A. Brett, G. Vincent, M. Orbes-Arteaga, D. Cárdenas-Peña, G. Castellanos-Dominguez, N. Aghdasi, Y. Li, A. Berens, K. Moe, B. Hannaford, R. Schubert, and K. D. Fritscher. "Evaluation of segmentation methods on head and neck CT: Auto-segmentation challenge 2015". In: *Medical Physics* 44.5 (2017), pp. 2020–2036. doi: https://doi.org/10.1002/mp.12197. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.12197. url: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12197.

[32] K. Jensen, J. Friborg, C. R. Hansen, E. Samsøe, J. Johansen, M. Andersen, B. Smulders, E. Andersen, M. S. Nielsen, J. G. Eriksen, J. B. B. Petersen, U. V. Elstrøm, A. I. Holm, M. Farhadi, M. H. Morthorst, P. S. Skyt, J. Overgaard, and C. Grau. "The Danish Head and Neck Cancer Group (DAHANCA) 2020 radiotherapy guidelines". In: *Radiotherapy and Oncology* 151 (Oct. 1, 2020), pp. 149–151. doi: 10.1016/j.radonc.2020.07.037. url: https://doi.org/10.1016/j.radonc.2020.07.037.

[33] G. Bansal, B. Nushi, E. Kamar, W. Lasecki, D. Weld, and E. Horvitz. "Beyond Accuracy: The Role of Mental Models in Human-AI Team Performance". In: *HCOMP 2019*. AAAI. Oct. 2019. url: https://www.microsoft.com/en-us/research/publication/beyond-accuracy-the-role-of-mental-models-in-human-ai-team-performance/.

[34] B. McCrindle, K. Zukotynski, T. E. Doyle, and M. D. Noseworthy. "A Radiology-focused Review of Predictive Uncertainty for AI Interpretability in Computer-assisted Segmentation". In: *Radiology: Artificial Intelligence* 3.6 (2021), e210031. doi: 10.1148/ryai.2021210031. eprint: https://doi.org/10.1148/ryai.2021210031. url: https://doi.org/10.1148/ryai.2021210031.

[35] G. Bansal, B. Nushi, E. Kamar, D. S. Weld, W. S. Lasecki, and E. Horvitz. "Updates in Human-AI Teams: Understanding and Addressing the Performance/Compatibility Tradeoff". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 33.01 (July 2019), pp. 2429–2437. doi: 10.1609/aaai.v33i01.33012429. url: https://ojs.aaai.org/index.php/AAAI/article/view/4087.

[36] M. Musleh, L. P. Muren, L. Toussaint, A. Vestergaard, E. Gröller, and R. G. Raidou. "Uncertainty guidance in proton therapy planning visualization". In: *Computers & Graphics* 111 (2023), pp. 166–179.

[37] C. E. Cardenas, B. M. Beadle, A. S. Garden, H. D. Skinner, J. Yang, D. J. Rhee, R. E. McCarroll, T. J. Netherton, S. S. Gay, L. Zhang, and L. E. Court. "Generating High-Quality Lymph Node Clinical Target Volumes for Head and Neck Cancer Radiation Therapy Using a Fully Automated Deep Learning-Based Approach". In: *International Journal of Radiation Oncology*Biology*Physics* 109.3 (2021), pp. 801–812.

[38] J. M. Wolfe, E. M. Palmer, and T. S. Horowitz. "Reaction time distributions constrain models of visual search". In: *Vision research* 50.14 (2010), pp. 1304–1311.

4

INCLUSION DEPTH FOR CONTOUR ENSEMBLES

In the previous chapters, we analyzed the clinical workflow. In the remainder of the dissertation, we focus on how to support the visual analysis of segmentation ensembles analysis, which directly impacts clinicians' efficacy in the quality assessment process. In this chapter, we introduce Inclusion Depth (ID), a novel method for analyzing segmentation ensembles by summarizing their key statistical features, such as median contours, confidence intervals, and outliers. ID overcomes the computational challenges of traditional contour depth methods by leveraging inside/outside relationships between contours, enabling efficient analysis of large ensembles. This approach allows for intuitive and rapid exploration of ensemble characteristics, making it potentially valuable for clinical quality assessment workflows in radiotherapy. By focusing on scalability, theoretical soundness, and ease of implementation, this chapter lays the foundation for advancements in contour depth computation and visualization techniques developed in Chapter 5.

This chapter is based on the published paper: Chaves-de-Plaza, Nicolas F., P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "Inclusion Depth for Contour Ensembles". In: IEEE Transactions on Visualization and Computer Graphics 30.9 (2024), pp. 6560–6571. doi: 10.1109/TVCG.2024.3350076

4.1. INTRODUCTION

Different applications in simulation, computer-aided design, and semantic segmentation have to deal with ensembles of curves. Analyzing these ensembles permits understanding uncertainties in the results. We focus on ensembles of spatiotemporal scalar fields from which one can extract contours, closed and consistently-oriented curves. These appear in several domains. One example is meteorology, where analysts use ensembles of weather forecasts to analyze the predictions' variability under different initial conditions or changes in the computational model [1]. Another example is semantic segmentation, where ensembles are used to quantify the uncertainty that might come from the training data or the model [2]. In image-guided medical specialties, ensembles of segmentations are analyzed for planning the patients' treatments [3].

Visual inspection of the ensemble can facilitate its analysis and understanding. Spaghetti plots, which draw each contour in the ensemble using a different color, are a popular technique. They are attractive because they are accessible, represent all the data, and are simple to implement. Nevertheless, as the ensemble size increases, spaghetti plots become cluttered, potentially hiding interesting features of the ensemble. Motivated by these limitations, ensemble summarization methods have been proposed. They reduce information by extracting features of interest, such as representative members and contour variability, from the ensemble and visualize them using visual encodings based on lines and bands [4–7].

A successful contour summarization technique is the contour boxplot (CBP) [7], which has been used in the fields of meteorology [8] and medicine [9–11]. As Figure 4.1 illustrates, like traditional boxplots, CBPs depict four statistical features of an ensemble: the median, the trimmed mean, confidence intervals, and outliers. Underlying the CBP is the concept of statistical depth, which extends univariate order and rank statistics to complex multivariate datasets by establishing a center-outward measure of centrality for the ensemble members[12].

In this paper, we propose an alternative notion of contour depth called Inclusion Depth (ID). ID contributes to the arsenal of depth-based contour analysis methods in three ways.

First, ID provides a novel statistical depth for ensembles of contours. It draws inspiration from Half-Region Depth (HRD) [13] and generalizes HRD from the class of functions to contours. This connection to HRD endows ID with theoretical properties and enables computational advantages analogous to those of HRD, also for ensembles of contours. In Section 4.4, we present the ID framework, detailing how it overcomes the challenges that extending the HRD approach to the contour case brings.

Second, ID leverages a simple principle that makes it accessible and facilitates the interpretation of the results. Specifically, ID leverages the inside/outside relationships between contours to estimate the ensemble's depth. To compute a contour's ID we compute how many other contours of the ensemble the contour contains and in how many other contours it is contained. Intuitively, a highly central contour has similar values for both quantities. An outlier might have an asymmetry of these quantities, if it's a magnitude outlier, or lower values for both, in the case of a shape outlier.

Third, the computation of ID scales better than the state-of-the-art Contour Band Depth (CBD), which was introduced jointly with the CBP idiom [7]. As Section 4.3 shows, for a N-contour ensemble, ID requires $\mathcal{O}(MN^2)$ operations while CBD needs at least $\mathcal{O}(MN^3)$, where M is the contour size (i.e., resolution of the binary mask). In Section 4.6, we evaluate ID, empirically showing that performing only pairwise comparisons does not degrade ID's performance and yields depth scores qualitatively comparable to CBD's.

We further demonstrate the practical use of ID in Section 4.7 by performing depth-based exploratory analysis of several real datasets from diverse domains like segmentation in radiotherapy and meteorological forecasting. Based on the results, we expect the faster but still performant ID will enable visual analysis of larger ensembles using depth-based visualizations like CBP, which allows both quantitative and qualitative interpretation of contour ensembles. Furthermore, it will bring applications that require multiple or/and fast depth evaluations like regression [14] and clustering [15] within reach.

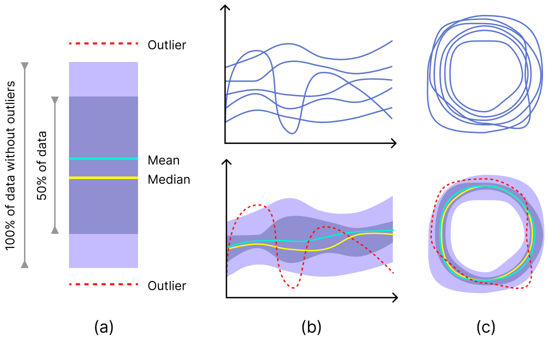


Figure 4.1.: Extension of the boxplot idiom (a) to the functional (b) and contour (c) data types.

4.2. RELATED WORK

Our method fits in the context of uncertainty visualization. Ensembles permit quantifying predictive uncertainties due to changes in the initial conditions, the training data, or the model parameters [2]. Existing ensemble visualization techniques can be cataloged based on their data type, visualization method, and analytic task [16].

There are several alternatives for presenting a visual overview of contour ensembles. Spaghetti plots are a composition-after-visualization technique that plots each contour using a different color [17]. Although straightforward to implement and interpret, spaghetti plots become cluttered as the size of the ensemble grows, potentially hiding trends and interesting members. To address this issue, several ensemble summarization techniques have been proposed in recent years that aggregate contour data into salient features before visualizing it. Most available summarization techniques share a visual language that uses contour lines for the ensembles' representative members like the median, mean, and outliers, and bands for areas of interest like the ensemble's spread [18] and confidence intervals [7].

Available summarization techniques differ in the features they compute and the assumptions they make. Parametric model-based techniques assume a data distribution and use available models to derive statistical quantities. One approach fits a Gaussian distribution on the contours' PCA-reduced signed distance field (SDF) transform and uses it to derive a median and calculate bands [5, 19]. It is also possible to use a Gaussian model to describe each grid point and use this model together with iso-contour density and level-cross probability to extract the iso-contours' probability density [20]. Parametric techniques are conceptually attractive as they permit extracting information analytically [5]. Nevertheless, they impose assumptions on the data, like normality, which limits the applicability in practice. Our method is fully non-parametric, using a depth-induced ranking of the contours to detect outliers and derive quantities of interest like the median and robust mean.

The family of data-based non-parametric methods does not impose assumptions on the data distribution and, therefore, can describe the ensemble data on each point more accurately [21, 22]. Local summarization methods operate on the grid in which contours lie, computing point-wise statistics. Examples are contour probability plots, which extract bands by thresholding a scalar field of percentages [18], and EnConVis [4], which performs point-wise kernel density estimation, and then uses the per-point density to extract bands and representatives. Contour grid points are not independent of each other, so computing summaries based solely on point-wise estimates can fail to consider global characteristics of the contour data like the topological relationships between contours.

The method introduced in [6] uses a vector-to-closest-point representation along the contours boundary points to quantify their centrality based on the vector lengths and directions. This approach requires only comparisons between contours, making it more efficient than CBD. Nevertheless, it uses parametric statistical models that require parameter fitting to obtain the centrality estimates. Furthermore, it is unclear how the method performs under different ensemble distributions, which makes it hard to compare to existing contour depth methods like CBD.

4.3. BACKGROUND: CONTOUR DEPTH AND BOXPLOTS

4.3.1. STATISTICAL DEPTH

Statistical depth provides a framework for extending concepts like the median, trimmed mean, and outliers, which depend on the points' ranks and orderings from the univariate to the multivariate case. Given a cloud of N d-dimensional points $X \in \mathbb{R}^{N\times d}$, a depth function $D(z,X) : \mathbb{R}^d \to [0,1]$ yields a center-outward measure of the centrality or depth of a point z with respect to X. Intuitively, the farther away a point z is from the center of X, the lower its centrality. In practice, there are different methods for computing $D(z,X)$, which come with different guarantees in terms of the function's behavior like invariance to different geometric transformations of X [12].

Statistical depth functions were originally devised to handle multivariate data. Nevertheless, their performance might decrease when $d \gg N$ due to the curse of dimensionality [23]. Furthermore, in some cases, data is more naturally represented as functions. In response to these observations, several definitions of depth that apply to functional data have been recently proposed [13, 23]. Two predominant functional depth methods are Band Depth (BD) [23] and Half-Region Depth (HRD) [13]. Inspired by the multivariate simplicial depth [24], BD computes a function's depth by comparing it to the bands formed by all other subsets of functions in the ensemble. Contour Band Depth, presented in the next subsection, generalizes BD's formulation and extends it to the case of contours.

Instead of forming bands, HRD looks at the proportion of functions lying on each side of the function of interest to determine its depth. The multivariate analog of HRD is Tukey's half-space depth [25]. HRD is more computationally efficient than BD, requiring only N comparisons per function. Furthermore, it has been shown to yield comparable depths to BD [13]. The proposed Inclusion Depth generalizes HRD's formulation and extends it to the case of contours. In the following, we outline HRD.

Let $X = \{x_1, x_2, \dots, x_N\}$ with $x_i : I \to \mathbb{R}$ be an ensemble of functions defined on the compact interval I. The graph of a function $x \in X$ can be defined as

$$G(x) = \{(t, x(t)), t \in I\} \tag{4.1}$$

The epi and hypographs of x, which correspond to the regions above and below $G(x)$, can be defined as

$$\begin{aligned} hyp(x) &= \{(t, y) \in I \times \mathbb{R} : y \leq x(t)\}, \\ epi(x) &= \{(t, y) \in I \times \mathbb{R} : y \geq x(t)\}. \end{aligned} \tag{4.2}$$

The HRD of x can be computed by evaluating the proportion of times $G(x)$ is contained in the epi and hypographs of other functions of the ensemble. Formally,

$$\mathrm{HRD}(x|X) = \min\{\mathrm{IN}_{hyp}(x), \mathrm{IN}_{epi}(x)\}, \tag{4.3}$$

where

$$\begin{aligned} \mathrm{IN}_{hyp}(x) &= \frac{1}{N}\sum_{i=1}^{N} G(x) \subset hyp(x_i), \\ \mathrm{IN}_{epi}(x) &= \frac{1}{N}\sum_{i=1}^{N} G(x) \subset epi(x_i), \end{aligned} \tag{4.4}$$

where $A \subset B$ is 1 if A is contained in B and 0 otherwise.

HRD in Eq. 4.3 attains its maximum value of 0.5 when $G(x)$ is contained in as many epi and hypographs of the other functions in the ensemble. The HRD satisfies several of the properties of a valid depth function [1]: linear invariance, maximality at the center, monotonically decreasing on rays, and upper-semicontinuity. Finally, a finite-dimensional version can be obtained by drawing d samples from I. When $d = 1$, the Half-Region Depth is equivalent to the Tukey depth.

4.3.2. CONTOUR BAND DEPTH

Statistical depth allows for robust and model-free exploratory data analysis. Contour Band Depth (CBD) permits applying the depth methodology to contours [7]. Similarly to functional BD, CBD computes a contour's depth by determining how many bands formed by all other possible J-sized contour subsets (where $J \in \mathbb{Z}$ and $J \geq 2$) contain the contour. A contour is in a band if it contains the intersection of the band's contours and is contained in their union. To reduce the computational cost of verifying contour containment in $\sum_{i=2}^{N}\binom{N}{i}$ bands (where N is the size of the contour ensemble), $J = 2$ is used. To alleviate the tendency of CBD with $J = 2$ to produce depth ties, a modified CBD (mCBD) was proposed [7]. Instead of strictly enforcing the containment property, mCBD considers the proportion of the contour that falls outside each band when computing its depth. CBD and mCBD compute an ensemble's depths in $\mathcal{O}(MN^3)$ time, where M is the contour size (i.e., binary mask resolution).

4.3.3. CONTOUR BOXPLOTS

Boxplots offer a visualization of a dataset's summary statistics. Specifically, as Figure 4.1 illustrates, a boxplot has four components. The gold and blue-colored lines represent the median and the trimmed mean, respectively. The trimmed mean is the average of the dataset with the outliers removed. Purple bands around the mean encode the interquartile range. Finally, outliers are shown using red dashed lines. As the middle and right side of Figure 4.1 shows, the idea of boxplots can be extended to ensembles of functional [26] and contour [7] types through the concept of functional and contour depth. In these cases, the per-member depth values are used to compute the different statistics. The median is the member with the highest depth value and the interquartile ranges are bands formed by members whose depths fall in the specified ranges. Finally, the members with the lowest depths are flagged as outliers.

4.4. INCLUSION DEPTH

In this section, we introduce Inclusion Depth (ID). While ID can be defined for contours in $\mathbb{R}^2$ and $\mathbb{R}^3$, for the sake of simplicity, we consider the two-dimensional case.

Let $C = \{c_1, c_2, \ldots, c_N\}$ be an ensemble of contours, where a contour c_i is a pair of a function $F_i : \Omega \to \mathbb{R}$ and an isovalue $q_i \in \mathbb{R}$. Here Ω is a compact domain in $\mathbb{R}^2$, such as a rectangle. A contour encloses a subset in the plane that we call the inside region:

$$in(c_i) = \{p \in \Omega | F_i(p) < q_i\}. \tag{4.5}$$

ID is based on a simple principle. We evaluate for all pairs $c_i, c_j \in C$ whether or not $in(c_i)$ is contained in $in(c_j)$. Then, we form the fraction of contours of C in which $in(c_i)$ is contained,

$$\mathrm{IN}_{in}(c_i) = \frac{1}{N}\sum_{j=1}^{N} in(c_i) \subset in(c_j), \tag{4.6}$$

and the fraction of contours of C that are contained $in(c_i)$,

$$\mathrm{IN}_{out}(c_i) = \frac{1}{N}\sum_{j=1}^{N} in(c_j) \subset in(c_i). \tag{4.7}$$

In the sums in Eqs. 4.6 and 4.7, we interpret $in(c_i) \subset in(c_j)$ as the numerical value 1 if $in(c_i)$ is contained in $in(c_j)$ and as 0 otherwise. The ID is the minimum of the two fractions

$$\mathrm{ID}(c_i|C) = \min\{\mathrm{IN}_{in}(c_i), \mathrm{IN}_{out}(c_i)\}. \tag{4.8}$$

Figure 4.2 illustrates the process of computing a contour's ID in a four-contour ensemble. As (a) depicts, ID is related to HRD. Specifically, the proof sketch in the appendix shows that if there is an invertible transform mapping the contours to graphs of functions, our definition of ID is the same as HRD. (b) presents the four comparisons required to compute IN$_{in}$ and IN$_{out}$ in Eq. 4.8. Note that the example uses perfectly nested simple single-loop contours for illustrative purposes. In practice, contours often have more complex shapes, are not necessarily nested, and can have multiple connected components.

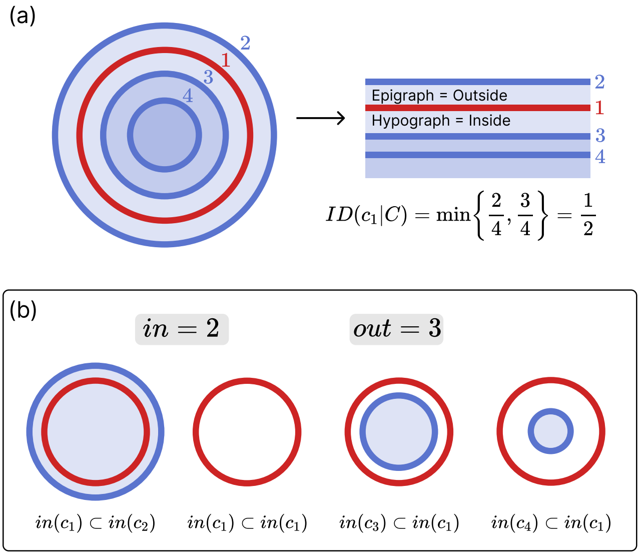

(a)
2
1
3
4
Epigraph = Outside
Hypograph = Inside
2
1
3
4
$ID(c_1|C) = \min\left\{\frac{2}{4}, \frac{3}{4}\right\} = \frac{1}{2}$
(b)
$in = 2$
$out = 3$
$in(c_1) \subset in(c_2)$
$in(c_1) \subset in(c_1)$
$in(c_3) \subset in(c_1)$
$in(c_4) \subset in(c_1)$

Figure 4.2.: Example of the ID computation for a 4-contour ensemble. In red is the contour for which we are currently estimating the depth. (b) shows the four comparisons that need to be performed to compute ID based on Eq. 4.8. Note that $c1 \subset c1$ (second column) counts for the inside and outside relationships.

ID is more general than HRD, accommodating the different topologies that arise in higher dimensions. Figure 4.3 shows examples of how ID deals with different cases. Note that, by subset operations, the definitions of IN$_{in}(c)$ and IN$_{out}(c)$ in Eqs. 4.6 and 4.7 ensure that the two contours under comparison are nested. As the bottom right panel of Figure 4.3 shows, when contours are not nested, the comparison will not add to the inside or outside counts, effectively reducing the depth of the contour under consideration. ID's results are invariant to homeomorphic transformations of the domain Ω, a general class of transformations that includes affine transformations and Möbius transformations. Additionally, ID's results are invariant to the choice of inside and outside. We sketch proofs of these properties in the appendix and point the interested reader to the set theory literature, which further elaborates on properties of the $\subset$ operator like transitivity [27].

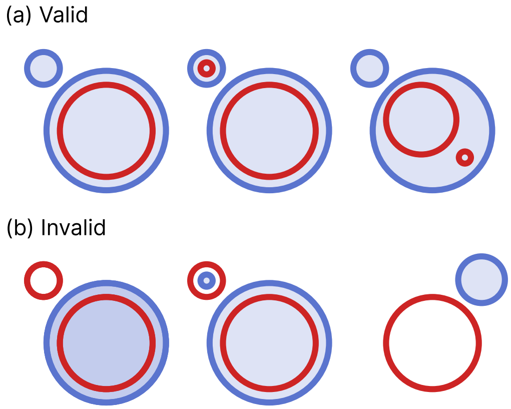


Figure 4.3.: Examples of how ID deals with different cases. If contours are nested (a), their relationship will add to the inside/outside counts. In other cases (b), the inside/outside counters will not increase, effectively reducing the overall depth.

Algorithm 1 shows how to compute the ID of a contour ensemble. For computations, we assume Ω to be a rectangle, *e.g.*, the bounding box of the ensemble of contours, and discretize the rectangle by a regular grid of size M. ID's scaling behavior depends mainly on the ensemble's size (N). Nevertheless, the grid size will also impact the algorithm's scaling behavior when performing the inside/outside comparisons. Therefore, ID has a computational complexity of $\mathcal{O}(MN^2)$, which is a significant improvement over the $\mathcal{O}(MN^3)$ complexity of CBD.

4.5. EPSILON INCLUSION DEPTH

If the ensemble's contours are non-smooth and intersect, like the examples in the center of Figure 4.4 show, inside/outside relationships will be ambiguous. In these cases, ID will produce ties and low-depth scores that reduce the method's practical utility. In this section, we present the Epsilon Inclusion Depth (eID) that relaxes the definitions of inside/outside in ID, reducing the effect of highly varying contours on the depth estimate.

For this extension, we proceed analogously to HRD, for which modified

Algorithm 1 Inclusion Depth (ID)

```
Require: $C, N$                                  ▷ Contour ensemble, number of contours
  $\mathbf{d}^{\mathrm{ID}} \leftarrow \{\}$                                   ▷ Inclusion depths
  for $i = 1$ to $N$ do
      $num\_in \leftarrow 0; num\_out \leftarrow 0$                   ▷ Inside/outside counts
      for $j = 1$ to $N$ do
          $num\_in \leftarrow num\_in + [in(c_i) \subset in(c_j)]$
          $num\_out \leftarrow num\_out + [in(c_j) \subset in(c_i)]$
      end for
      $\mathrm{IN}_{in}(c_i) = num\_in/N$
      $\mathrm{IN}_{out}(c_i) = num\_out/N$
      $\mathbf{d}^{\mathrm{ID}} \leftarrow \mathbf{d}^{\mathrm{ID}} \bigcup \min\{\mathrm{IN}_{in}(c_i), \mathrm{IN}_{out}(c_i)\}$
  end for
  return $\mathbf{d}^{\mathrm{ID}}$
```

HRD (mHRD) alleviates the problem that strongly varying functions pose for HRD by relaxing the requirement that the graph of a function must lie entirely in the epi or hypograph. mHRD determines the average proportion of the domain that a function's graph lies in the hypo and epigraphs of other functions [13]. This strategy is not directly applicable to the case of contours because of the lack of an independent variable. Therefore, we follow a strategy inspired by the modified Contour Band Depth in [7], which operates directly on the contours' domain and therefore does not require a dependent variable.

First, we define the epsilon subset operator $A \subset_\epsilon B$ for two sets $A, B \subset \mathbb{R}^2$. In contrast to the subset operator $\subset$, which returns either 0 or 1, $\subset_\epsilon$ yields a value in the interval $[0, 1]$. It is defined as

$$A \subset_\epsilon B = 1 - \begin{cases} 0 & |A| = 0, \\ |A - B|/|A| & \text{otherwise,} \end{cases} \tag{4.9}$$

where $|A|$ denotes the area of A and $A - B$ the set difference. Note that $A \subset_\epsilon B$ will be one if B contains A. If a part of A lies outside of B, $\subset_\epsilon$ will yield lower values.

The definition of eID is analogous to ID except that the $\subset$ operator is replaced by the $\subset_\epsilon$ operator. We consider the values

$$\begin{aligned} \mathrm{IN}^{\epsilon}_{in}(c_i) &= \frac{1}{N}\sum_{j=1}^{N} in(c_i) \subset_\epsilon in(c_j), \\ \mathrm{IN}^{\epsilon}_{out}(c_i) &= \frac{1}{N}\sum_{j=1}^{N} in(c_j) \subset_\epsilon in(c_i). \end{aligned} \tag{4.10}$$

The eID is the minimum of the two values

$$\epsilon\mathrm{ID}(c|C) = \min\{IN^{\epsilon}_{in}(c), IN^{\epsilon}_{out}(c)\}. \tag{4.11}$$

Figure 4.4 shows how $\subset_\epsilon$ works across a variety of cases. As the extremes of the first row illustrate, when $in(c_i)$ (red) is completely inside or outside of c_j (blue), the difference between $in(c_i) \subset_\epsilon in(c_j)$ and $in(c_j) \subset_\epsilon in(c_i)$ is the largest. When the relationship between the contours is ambiguous, the second row of the figure shows that the difference shrinks. Also, the values of these quantities decrease, which has the effect of reducing the contribution of the c_i/c_j comparison to the overall depth calculation. Finally, eID is invariant to area-preserving transformations. We sketch the proof of this property in the appendix.

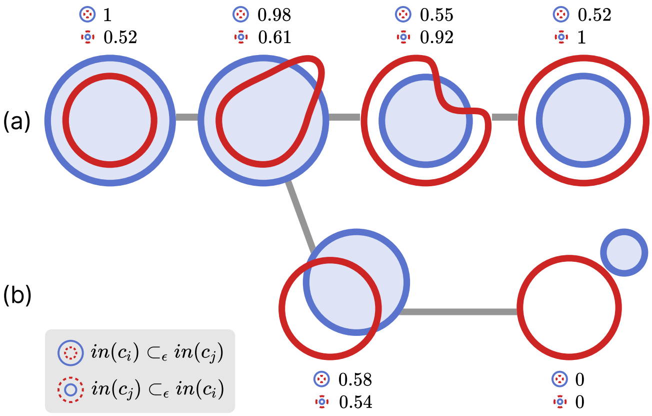

1
0.52
0.98
0.61
0.55
0.92
0.52
1
(a)
(b)
$in(c_i) \subset_\epsilon in(c_j)$
$in(c_j) \subset_\epsilon in(c_i)$
0.58
0.54
0
0

Figure 4.4.: Examples of computing the inside and outside relationships with the $\subset_\epsilon$ operator in Equation 4.9 for different contour configurations. In red and blue are contour $\{c_i, c_j \in C\}$. The first row shows the transition of c_i from being completely inside to completely outside of c_j. The second row shows the values that $\subset \epsilon$ yields in ambiguous cases.

As the next sections show, eID provides meaningful results even when contours have many intersections. The implementation of eID only requires swapping $\subset$ for $\subset_\epsilon$ in Algorithm 1. eID maintains ID's computational complexity of $\mathcal{O}(MN^2)$.

4.6. EXPERIMENTS

In this section, we perform an extensive evaluation of the Inclusion Depth (ID) method using synthetic data. Specifically, we assess the scaling behavior of ID as the dataset's size increases and investigate the robustness of estimators derived with ID and the method's performance at identifying outliers. Before continuing with the experiments, we detail our experimental setup.

4.6.1. EXPERIMENTAL SETUP

In our experiments, we compare ID and eID with Contour Band Depth (CBD) and its modified version (mCBD). CBD's only parameter, the number of contours forming the band (J), is set to J=2 for all experiments.

We implemented the CBD and ID methods and the experiments in a Python-based framework [1]. Contour depth methods receive as input a list of binary Numpy [28] arrays of size $M = 300 \times 300$ corresponding to a discretization of Eq. 4.5. These binary masks can be obtained, for example, as the output of a segmentation algorithm or by thresholding scalar fields using an iso-value. While acceleration through parallelization is possible, our focus in implementation prioritizes asymptotic algorithmic scaling over specific optimizations.

Similar to [7], we use synthetic ensembles of circular shapes contaminated with outliers to assess the methods' performance. We extend the experiments of contour depth by considering different types of outliers separately, following the experimental paradigm used to evaluate the functional Half-Region Depth [13]. The first row of Figure 4.5 showcases the different outliers we consider (orange contours). We expect the depth scores that CBD and ID yield to be lower for contours that deviate from the ensemble's main trend and higher for those that conform to it.

To generate ensembles of contours contaminated with outliers, we define a stochastic model from which we can sample shapes. The model results from a mixture of a base model r_0 and a second model r_1, which depends on the outlier type under consideration. For both r_0 and r_1, we use stochastic processes indexed by the shape's angle, yielding angle-correlated values for the shape's radius. We define the base model r_0 as

$$r_0(\theta) = f_0(\theta) + \epsilon_0(\theta), \tag{4.12}$$

where $\theta \in \mathbb{R}^1 00$ is a vector containing 100 equally spaced samples of the interval $[0, 2\pi]$ and $f_0(\theta) = 0.5$ is the mean radius function.

To add randomness to the mean shape, we use Gaussian Processes (GP), defined by a mean and an exponentiated quadratic kernel

$$k_{mid}(\theta_i, \theta_j) = \sigma_{mid}^2 \exp\left(-\frac{(g(\theta_i) - g(\theta_j))^2}{2l_{mid}^2}\right), \tag{4.13}$$

where $\theta_i, \theta_j \in \theta$, $g : \mathbb{R} \to \mathbb{R}$ is a function that transforms the domain and *mid* can be zero or one depending on whether we refer to r_0 or r_1 in Eq. 4.14.

We define $\epsilon_0(\theta)$ in Equation 4.12 as the sum of two zero-mean GPs with $g = \sin$ and $g = \cos$ in Eq. 4.13, respectively. Using these periodic functions ensures that the start and end of the θ interval are mapped

[1] Code can be found at https://graphics.tudelft.nl/inclusion-depth-paper

to the same radius. The kernel's parameters σ_0 and l_0 define the shape of the contour by affecting the amplitude and the frequency of the angle-correlated noise. We set $\sigma_0 = 0.003$ and $l_0 = 0.9$.

To obtain a binary mask from the zero-centered shape defined by the polar coordinates $(\theta,\ r(\theta))$, we convert them to Cartesian coordinates using $y = r\sin(\theta)$ and $x = r\cos(\theta)$, and rasterize the resulting closed polygon in a square grid with the target size M with scikit-image's polygon2mask. The panel in the upper left corner of Figure 4.5 shows a $N = 100$ ensemble generated by sampling the base model r_0 (D1).

For the experiments, we define five datasets of contour ensembles (D2-D6 in Figure 4.5) based on the three types of outliers we describe next. In all cases, we obtain an outlier-contaminated ensemble by sampling from the mixture

$$r(\theta) = r_0(\theta) + \rho r_1(\theta), \tag{4.14}$$

where $\rho \sim Bern(0.1)$ introduces an outlier with a probability of 0.1 and r_1 is defined analogously to r_0 in Equation 4.12. In the following, we describe the different outlier types.

First, we consider magnitude outliers in which we alter the shape's mean radius. We define the auxiliary random variable $sign = 2\gamma—1$ where $\gamma \sim Bern(0.5)$. $sign$ indicates whether the magnitude contamination corresponds to shrinking (-1) or enlarging (1) the shape. The first dataset with magnitude outliers is the Symmetric Magnitude Contamination (D2) for which $f_1(\theta) = 0.3 \cdot sign$. We define a second dataset with magnitude outliers which we call Peaks Magnitude Contamination (D3). Instead of changing the magnitude of the shape's radius, in D3 we only contaminate a subinterval (θ_l, θ_r) of θ where $\theta_l < \theta_r$ and both θ_l and θ_r are uniformly distributed random variables. Specifically, for D3, we define f_1 as

$$f_1(\theta) = \begin{cases} sign \cdot inc & \theta_l \leq \theta \leq \theta_r \\ 0 & \text{otherwise} \end{cases}$$

where $inc = 0.3$, and θ_l and θ_r are defined for every $\theta_i \in \theta$.

The second type of outlier we consider is shape outliers. To obtain shape outliers, instead of altering the mean radius of the circular shape, we modify the parameters of the covariance matrix of ϵ_1 which define the amplitude (σ_1) and the frequency (l_1) of the noise along the shape's boundary. Specifically, increasing σ_1 leads to higher amplitude while increasing l_1 increases the number of peaks. For the Shape Inside (D4) dataset, we keep $\sigma_1 = 0.003$ but decrease the frequency to $l_1 = 0.01$ to ensure that the shape varies while staying within the ensemble's envelope. For the Shape Outside (D5) dataset, we set $\sigma_1 = 0.009$ and $l_1 = 0.04$, which results in highly varying shapes that spill outside the bounds defined by the normal members of the ensemble. We expect D4 outliers to be more challenging to detect than D5 ones, given that they fall inside the ensemble's envelope.

The final type of outlier we consider are topological outliers which correspond to contours that have holes or disconnected components not present in other members of the ensemble. To create the Different Topologies dataset (D6), we randomly downscale r_1 using a uniform distribution between 0.1 and 0.2 for the scaling factor. Note that we use the same parameters for r_1 as for r_0. After determining the (x, y) coordinates of the shrank shape, we translate them to a random location that lies either inside or outside (with equal probability) of the mean circular shape defined by r_0.

For the experiments, we consider several ensemble sizes $N \in \{i * 10 : 1 \leq i \leq i_{\max}$, where $i_{\max} = 10$ for CBD and $i_{\max} = 30$ for ID. We compute 10 realizations of each dataset/size/depth method combination to establish the results' statistical significance. We ran all the experiments presented in this section on a Mac Book Pro (2022) with an M1 Pro processor (without GPU acceleration) and 32 GB RAM.

4.6.2. EXPERIMENT 1: SCALING BEHAVIOR

Figure 4.6 depicts the time in seconds that each depth method takes for ensembles of different sizes. For each size, we compute the mean and standard deviation across replications and datasets (D1-6). The first thing to note is that we only ran CBD methods until $N = 100$. After this point, the CBD method took too long to compute. In contrast, we considered ensembles up to size $N = 300$ for ID. The figure shows how ID and eID, with a computational complexity of $\mathcal{O}(MN^2)$, scale more favorably than CBD methods, which are $\mathcal{O}(MN^3)$.

In addition to the aggregated runtime, we investigated the time the preprocessing and depth calculation loop portions of each method take. Table 4.1 shows this information for D1 with $N = 100$. As the table shows, all methods spend most of their time in the depth calculation loop (t2). CBD methods take, on average, an order of magnitude more time than ID methods. The large standard deviations of CBD methods' timings are caused by outlier timings that arose likely due to other processes in the machine interfering with the experiment's process. Within each method family, the modified version takes more time because they require more operations than the strict versions. Finally, CBD methods have a larger preprocessing time (t1) than ID methods, which do not require preprocessing. This is specific to our implementation, which precomputes CBD's bands before starting the depth calculation loop.

4.6.3. EXPERIMENT 2: OUTLIER DETECTION

Depths can be used to perform robust statistical analysis by removing outliers, which are contours with low depth. For the second experiment, we evaluate ID's performance in identifying outliers in D2-D6 in Figure 4.5. Specifically, given a set of outliers $\mathcal{O}_m$ for a method m and

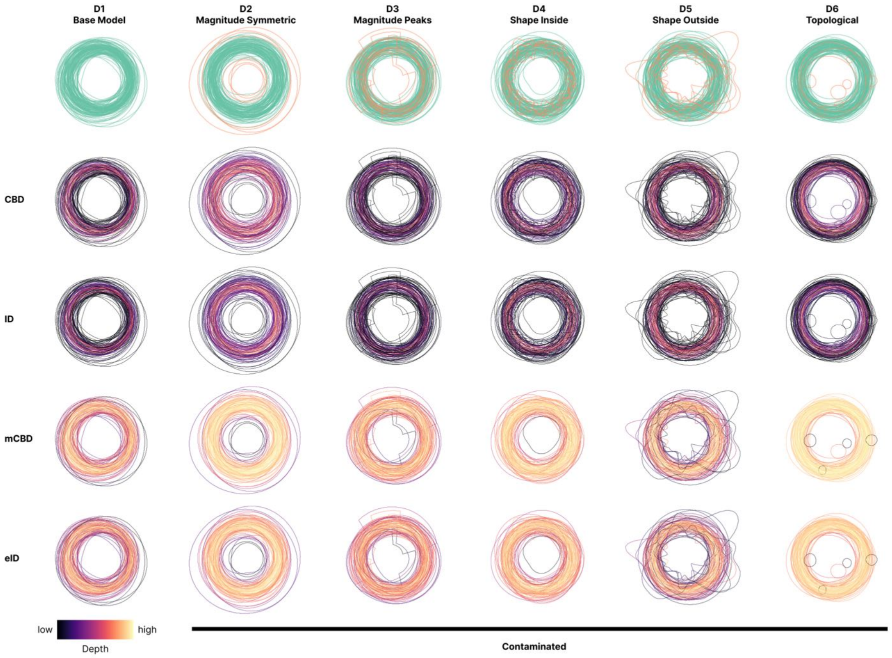


Figure 4.5.: The first row presents an overview of the synthetic datasets we used in the experiments, with the outliers highlighted in orange. The last four rows plot the ensembles assigning the lines' colors based on the depths each method yielded. Darker and brighter colors denote lower and higher depth values, respectively. The color scale was scaled based on the min and max depth value per dataset/depth method combination to facilitate the comparison of the depth-induced rankings across methods.

Table 4.1.: Mean and standard deviation of the preprocessing (t1), depth calculation loop (t2) and full (t3=t1+t2) times in seconds for D1 with $N = 100$.

Method	t1 (secs)	t2 (secs)	t3 (secs)
CBD	6.75 ± 1.77	612.31 ± 351.40	619.06 ± 351.14
mCBD	6.48 ± 1.46	697.02 ± 328.91	703.50 ± 328.49
ID	0.00 ± 0.00	2.31 ± 0.37	2.31 ± 0.37
eID	0.00 ± 0.00	7.37 ± 3.98	7.37 ± 3.98

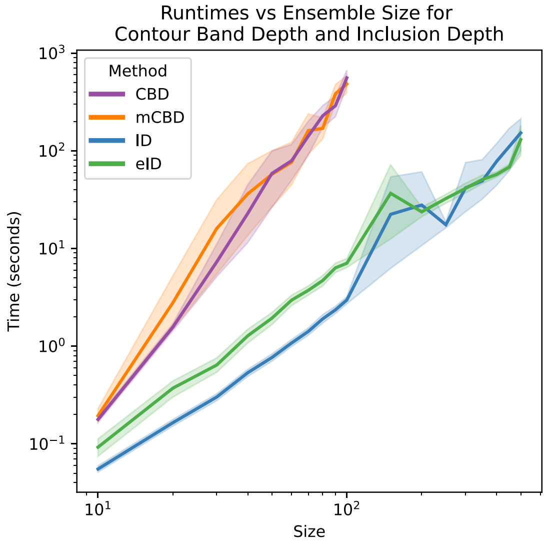


Figure 4.6.: Comparison of mean runtimes across datasets and replications of CBD, mCBD, ID and eID. Both x and y-axis use logarithmic scales and shaded area denotes the 95 percent confidence interval.

a reference set $\mathcal{O}_r$, we compute the percentage of correctly identified outliers with respect to the reference set as

$$PO_{m,r} = \begin{cases} 0 & \text{if } |O_r| = 0 \\ \frac{|O_m \cap O_r|}{|O_r|} & \text{otherwise,} \end{cases} \tag{4.15}$$

where $|\cdot|$ denotes the number of outliers in the set.

For a method m, we define its set of outliers O_m as the $\lceil N\alpha \rceil$ members with the lowest depths, where $\lceil \cdot \rceil$ is the ceiling operator. For the results we report next, we used $\alpha = 0.3$. We compare the outliers of each depth method identified against the ground truth (GT) outliers, which we define as the reference set O_r. Table 4.2 shows the mean and the standard deviation of the percentage of the outliers each method detected with respect to the GT ones for D2-D6 with $N = 100$.

As the table indicates, except for D2, strict depth methods are more effective at identifying outliers. This result agrees with the functional depth literature, which shows that strict depth methods have a higher sensitivity to outliers [23]. The most challenging dataset for mCBD and eID was D4, with inside-shape outliers. Although both methods performed poorly, mCBD did a better job, which potentially indicates that the extra comparisons of CBD endow the method with a higher sensitivity for detecting shape outliers.

As the table indicates, no strict method consistently outperforms the other. ID performed better for the dataset with symmetric magnitude contamination (D2) and topological outliers (D6). In the other cases, CBD achieved better scores. Similarly, except for D4, the performance of modified depth methods was comparable across datasets. These results show how, in practice, the choice of method will depend on the type of data at hand. In agreement with previous literature in band depths [7, 23], the strength of CBD lies in identifying outliers like those in D4, which have a significantly different shape but fall within the ensemble's band envelope.

Finally, we also compare the methods' outlier detection performance qualitatively. The four bottom rows of Figure 4.5 present the spaghetti plots with lines colored according to the depths that different methods yield. The figure evidences the similarities between CBD and ID, and mCBD and eID. As expected, CBD and ID methods assign lower depth values to contours that deviate from the ensemble's main trend. CBD and ID produce a wider range of depth values, demonstrated by the color gradient which contains black and bright yellow lines. In contrast, mCBD and eID yield mostly high-depth scores with some contours receiving lower ones. Graphically, this translates to overall brighter color gradients. Despite this visual change, it is possible to observe that the depth-induced rankings of the contours are similar between strict and modified versions.

Table 4.2.: Average percentage of outliers that Contour Band Depth (CBD) and Inclusion Depth (ID) methods detected with respect to the ground truth outliers for $N = 100$.

	CBD (%)	mCBD (%)	ID (%)	eID (%)
Dataset				
D2	76.16 ± 13.31	98.12 ± 4.22	90.08 ± 7.68	98.12 ± 4.22
D3	77.54 ± 14.67	58.94 ± 13.19	71.46 ± 14.48	49.89 ± 14.20
D4	88.14 ± 15.59	17.43 ± 14.74	85.07 ± 14.34	8.06 ± 7.68
D5	85.21 ± 16.92	69.27 ± 8.94	83.37 ± 18.90	54.54 ± 7.78
D6	66.11 ± 9.31	68.19 ± 16.23	81.46 ± 13.33	66.52 ± 18.90

4.6.4. EXPERIMENT 3: ESTIMATOR'S ROBUSTNESS

Depth values permit generalizing uni-variate order and rank statistics to the multivariate case. For this experiment, we are interested in the quality of the trimmed mean, which is one of the robust statistics that the contour boxplot visualization uses. To compute the α-trimmed mean (M_m^α) of an ensemble of contours we average binary masks of the top $N - \lceil N\alpha \rceil$ contours, depth-wise, and extract a new contour from the resulting scalar field using 0.5 as iso-value. Specifically, we compute the α-trimmed mean contour for method m using the expression

$$\mathrm{M}_m^\alpha = \frac{\sum_{i=1}^{N-\lceil N\alpha \rceil} in(c_i)}{N-\lceil N\alpha \rceil}, \tag{4.16}$$

where $in(c_1),\dots,in(c_{N-\lceil N\alpha \rceil})$ are the binary masks of the inside regions associated with the $N-\lceil N\alpha \rceil$ contours with the highest depth, according to method m. In addition to each method's trimmed mean, we also consider the sample mean (M_S), which we compute per dataset/replication combination by using Eq. 4.16 without trimming the ensemble. M_S represents a worst-case scenario in which outliers were not removed. For the experiments in this section, we set $\alpha = 0.3$.

A robust trimmed mean is one not affected by outliers. In other words, the trimmed mean contour should be close to the population's average shape. Therefore, to evaluate the depth methods' estimators, we compare them against the binary mask of f_0 in Eq. 4.12, which we denote M_P. To compare the trimmed means with M_P we compute the mean squared error (MSE) between the masks

$$MSE(M_m^\alpha, M_P^\alpha) = \frac{\sum_{r=0}^{rows}\sum_{c=0}^{cols}[M_m^\alpha(r,c) - M_P^\alpha(r,c)]^2}{rows \times cols}, \tag{4.17}$$

where $M_m^\alpha(r,c)$ is the value of the binary array of the trimmed mean M_i^α under consideration at the given row and column.

Table 4.3 presents the mean and the standard deviation of the MSE for D1-D6 with the ensemble size $N = 100$. Both CBD and ID methods yield lower average MSE when compared to the sample mean M_S. This shows that removing outliers, only considering the most central contours, leads to more robust estimators closer to the population mean M_P. In most cases, the mean MSE of M_α^{CBD} is higher than that of M_α^{ID}. The same observation holds for the modified versions, which suggests that the outliers ID methods remove contribute more to deviating the trimmed mean from the population estimate. Finally, modified depth methods obtain lower MSE than their strict counterparts. Considering that strict methods performed better at identifying outliers, this result suggests that other contours besides artificially introduced outliers might contribute more towards making the mean estimates less robust. These results show that both CBD and ID methods yield robust mean estimates that are closer to the population estimate than M_S.

4.7. VISUAL COMPARISON ON REAL DATA

The previous results demonstrated ID's robustness and more favorable scaling behavior compared to CBD using synthetic data. We now

Table 4.3.: Average MSE between population estimate M_P, and the sample mean (M_S) and alpha-trimmed means obtained with CBD (M^{α}_{CBD}), mCBD (M^{α}_{mCBD}), ID (M^{α}_{ID}) and eID (M^{α}_{eID}) depths. We compute the average MSE across replications for $N = 100$ and include also the standard deviation of the estimates. We multiply both the mean and std by $\times 10^2$ to reduce clutter.

Dataset	M_S	M^{α}_{CBD}	M^{α}_{mCBD}	M^{α}_{ID}	M^{α}_{eID}
D1	1.42 ± 0.06	1.17 ± 0.10	1.13 ± 0.04	1.15 ± 0.08	1.12 ± 0.05
D2	1.77 ± 0.08	1.47 ± 0.12	1.32 ± 0.14	1.37 ± 0.14	1.31 ± 0.12
D3	1.51 ± 0.08	1.26 ± 0.11	1.20 ± 0.12	1.24 ± 0.10	1.18 ± 0.11
D4	1.46 ± 0.08	1.24 ± 0.09	1.14 ± 0.05	1.22 ± 0.08	1.13 ± 0.05
D5	1.50 ± 0.08	1.24 ± 0.10	1.17 ± 0.07	1.24 ± 0.10	1.17 ± 0.07
D6	1.60 ± 0.16	1.48 ± 0.23	1.17 ± 0.08	1.24 ± 0.14	1.15 ± 0.06

illustrate the use of ID with medical image semantic segmentation and meteorological forecasting datasets. The contours in these real datasets tend to cross over a lot. Therefore, we focus the analysis on eID, which yields more visually meaningful results in these cases. Unless stated otherwise, we used the same setup for the depth computation methods and ran the analyses in the same machine as in the experiments with synthetic data.

4.7.1. MEDICAL IMAGE SEGMENTATION ENSEMBLES

Data In image-guided medical specialties, clinicians use three-dimensional images of the patient's anatomy to plan the treatment. A core step of the treatment planning process is to segment anatomies of interest like malignancies and the organs-at-risk. With the advent of deep learning-based auto-contouring technologies, this step has been largely automated [29]. Nevertheless, clinicians still need to perform a quality assessment of the segmentations, which requires understanding the uncertainty in the predictions.

We consider the computerized tomography (CT) of a patient with head and neck cancer treated at HollandPTC between 2018 and 2020. The IRB approved the research protocol for the use of patient data in research, all patients signed an informed consent form. For the analysis, we focus on the brain stem and the parotid gland because these structures are not always clearly visible in CT, which can increase inter-clinician variability. In these cases, a visual statistical summary can help clinicians understand the range of predictions. We used a collection of 3D segmentation models based on the popular UNet architecture [30] to generate an ensemble of segmentation predictions of the right parotid gland. Specifically, we trained 30 models on different subsets of the training split of the dataset of the Head and Neck Auto Segmentation MICCAI Challenge [31], a technique known as bootstrapping in the machine learning community. The MICCAI dataset contains CT scans of patients with head and neck cancer with ground truth segmentations of nine organs at risk. To further augment the ensemble size, and the variability of the predictions, we trained each model using different learnable weight initializations. Using the resulting models to segment the parotid gland yields an ensemble of 120 scalar maps of per-voxel softmax probabilities. We extracted the contour ensemble that CBD and ID receive as input by thresholding these arrays with an iso-value of 0.8. For the results below, we computed the depths of the ensemble of contours in 2D 540 × 540 pixels slices of the right parotid gland and brain stem segmentation volumes.

Analysis The top row of Figure 4.7 visualizes the raw ensemble of contours of the brain stem and parotid gland using spaghetti plots. The variability in the contours of the two structures differs due to changes in visibility in the CT scans. The brain stem shows significantly more variability than the parotid gland, especially on the upper left side, where several contour lines go out of the way of the main shape. The overplotting in the spaghetti plots creates high-density areas that hint at the location of shape representatives. Conversely, one would expect that contours in less dense areas that deviate from the ensemble's main trend correspond to outliers.

Visual statistical summaries remove the need from presenting all ensemble members while still conveying relevant statistical features like the representative contours and the ensemble's variability. For each anatomical structure, Figure 4.7 presents contour boxplots generated with depths from the mCBD and eID, using $\alpha = 0.1$ for the trimming. The first thing to note is the different runtimes. For a $N = 100$ ensemble, mCBD took more than twenty minutes to compute the depths. In contrast, it took eID seconds. These results show that ID can support larger datasets without requiring special hardware, which increases its practical value.

In terms of the boxplot's statistical features, we start by analyzing the median, depicted as a yellow line. In both cases, the median that mCBD and eID yield is not the same contour. Nevertheless, the contours' shapes are visually similar. When we inspected the depth-induced rankings of the contours, we noticed both medians obtained high depth with both methods, but their ranks varied, which resulted in a different contour being displayed. For instance, for the parotid gland, mCBD's median had the 8th highest depth according to eID. Similarly, eIDs' median was ranked 8th according to mCBD. We observed a similar trend with the brain stem. The similarity of the rankings induced by mCBD and eID depths can be observed by comparing the method's trimmed

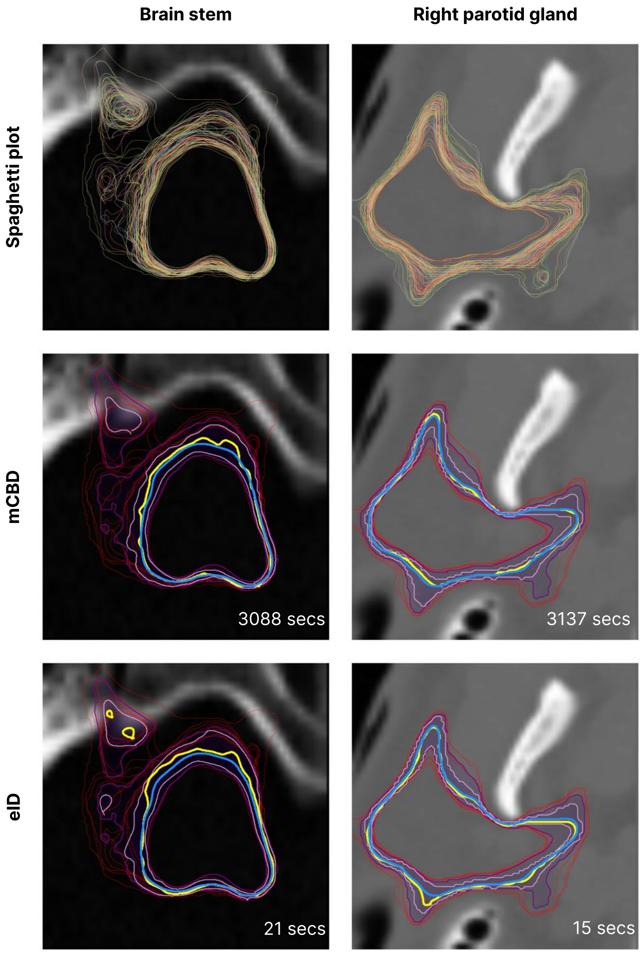


Figure 4.7.: Contour boxplots that provide a statistical summary of an ensemble of contours of a slice of the brain stem (top row) and right parotid gland (bottom row) of a head-and-neck cancer patient. We generated the contour boxplots using the depths obtained from the mCBD and eID. The yellow and blue lines correspond to the median and mean, respectively. Two bands are depicted in shades of purple as formed by members with the top 50% and 100% depths, not considering outliers, which are shown using dashed red lines. The timings indicate the duration of the contour depth computation process.

means (blue lines) and the outliers (dashed red lines). The figure shows how the means both methods yield are very similar. While the MSE between mCBD's and eID's median contours is 0.013 (brain stem) and 0.011 (parotid gland) pixels, it is 0.0005 (brain stem) and 0 (parotid gland) pixels for the means. In the appendix, we provide a juxtaposed view of the medians and α-trimmed means to facilitate visual inspection. The α-trimmed means are similar because mCBD's and eID's inlier and outlier sets largely agree. For the right parotid gland, for example, both sets perfectly intersect. Similar to the medians, the boxplots' confidence bands vary across methods. The reason is that although mCBD's and eID's inlier and outlier sets agree, the ordering of the contours is not the same (Pearson correlation coefficient between mCBD and eID depths of 0.96 for the brain stem and 0.98 for the parotid gland). Therefore, the shape of the bands might differ if, for example, a contour that fell in the 50% band for mCBD does not for eID.

4.7.2. METEOROLOGICAL FORECASTING

Data A common use case for contour statistical models is to analyze meteorological forecast data. In this work, we consider data from the European Centre for Medium-Range Weather Forecasts (ECMWF). Specifically, the ECMWF Ensemble Prediction System (EPS) provides ensembles of predictions for different variables like precipitation, temperature, and pressure. The forecasts include $N = 50$ perturbed members and a control run. We analyze the same data as in [5], which is the forecast from 00:00 UTC 15 October 2012. More details about this type of data can be found at [32]. The region under consideration encompasses $101 \times 41 \times 62$ grid points, which corresponds to latitude, longitude, and geopotential height dimensions. For the analysis, we consider 2D fields, corresponding slices of the region where the geopotential height is $500hPa$. To obtain contours from this field, we threshold them using an iso-value of 5600 m. The left-most panel of Figure 4.8, depicts the extracted contours laid over the geographical region they span.

Analysis The spaghetti plot in the first column of Figure 4.8 permits assessing the extent of the ensemble and suggests trends in high-density areas. Furthermore, it permits identifying contour portions lying outside of the ensemble's envelope as potential outliers. The second and third panels of Figure 4.8 color the lines using the depth that mCBD and eID assigned to each contour. Darker and brighter colors represent lower and higher values, respectively. Visual inspection reveals the similarity between the methods' outputs. Contours that are deep within the ensemble's predominant shape are brighter, signaling higher depth. In contrast, contours with portions deviating from the ensemble's

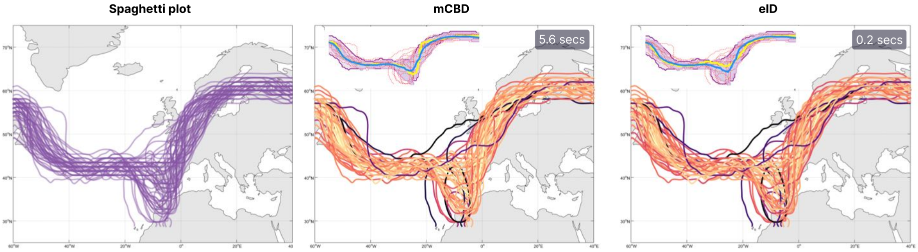


Figure 4.8.: mCBD and eID depths for an ensemble of 500 hPa geopotential height contour lines. The inset of each method presents the corresponding contour boxplot with the $N \times 20\% = 10$ contours with the lowest depth set as outliers. We used an opacity of 0.1 for the spaghetti plot. For the depth line plots, we scaled the color scale based on the min and max depth each method yielded. The timings indicate the duration of the contour depth computation process.

representative trend are darker. Additionally, the color gradients in both mCBD and eID line plots are similar. Inspecting the depth scores closer revealed a large agreement between the methods' inlier and outlier sets, which had 97% and 90% overlap (α=0.2). Nevertheless, similarly to the case of segmentation data, the specific depth values vary (correlation coefficient of 0.96), altering the depth-induced rankings and leading to different medians (yellow) and bands being displayed. The α-trimmed means (blue), with an MSE of 0.0021 pixels (compared to the medians' MSE of 0.037 pixels), and outliers (red) evidence the agreement of robust estimates based on mCBD and eID depth scores.

4.8. DISCUSSION AND CONCLUSION

In this paper, we presented Inclusion Depth (ID), a new depth notion applicable to contour ensembles. The concept of statistical depth permits extending order and rank-based statistics to the multivariate case. Depth-induced orderings allow summarizing the ensemble members in terms of their median, trimmed mean, and confidence bands, and obtaining robust estimators by removing outliers.

ID provides theoretical guarantees on the depth estimates, derived from its relationship with Half-Region Depth. Additionally, based on the simple principle of assessing contours inside/outside relationships, ID is accessible and its results interpretable. Using synthetic data we demonstrated ID's more favorable $\mathcal{O}(MN^2)$ scaling, compared CBD's $\mathcal{O}(MN^3)$ [7]. The experiments showed that ID and eID are successful at identifying a wide range of outliers and yield robust estimators of the ensemble's mean, comparable to CBD's. These robust estimators enable extending robust statistical theory and analysis to contours. Finally, by applying ID to real datasets, we further demonstrated the method's practical value to analyze contour ensembles when paired with visualizations like contour boxplots.

In the literature, it has been noted that CBD can be accelerated in several ways. First, CBD's outer loop is highly parallelizable, so it could significantly profit from GPU acceleration. In this paper, we did not focus on improvements that could be added on top of the methods. Rather, we propose an alternative depth notion that is asymptotically faster than CBD. Similarly to CBD, ID has a highly parallelizable loop, so this improvement would also benefit ID. Second, in terms of algorithmic improvements, [33] proposes a faster way to compute functional Band Depth. Contours, with the different possible topologies, are not straightforward to adapt to this methodology. Therefore, it remains future work to verify whether these optimizations are possible. Same as with parallelization, it holds that such an improvement would likely benefit both CBD and ID.

The experiments with synthetic data showed that ID and eID detect outliers with comparable performance to CBD across several outlier types. Nevertheless, there is still room for improvement. Particularly in the case of eID, which performed weakly at identifying shape outliers with a magnitude similar to other ensemble members. Improving outlying detection performance constitutes future work. We anticipate that introducing information about the contour's metric structure, similar to [6], could help in cases where inside/outside relationships do not suffice. Second, the eID can assign low non-zero depth scores to outlying contours. mCBD uses an automatic thresholding method that optimizes the ensemble's mean depth to set outliers' depth to zero. This procedure removes the need to find a threshold for the trimming operations via trial and error, like in eID's case. To reduce users' burden, we will investigate options to integrate an automated thresholding procedure similar to mCBD's in our framework.

The improved computational complexity of ID brings within reach the usage of depth-based order and rank statistics for larger datasets in interactive settings. In domains like computer-aided design, simulation, and medical image segmentation, it is common to deal with three-dimensional objects [11]. Our method is quite general and can be applied to three-dimensional contours with ease. Second, currently unimodal distribution is assumed, however, when studying contour's ensembles it is common to first identify the main modes of variation [4, 19, 34]. CBD could make this identification more robust to certain types of outliers [15] but at the cost of reduced interactivity. Using ID instead would permit performing real-time interactive depth-based clustering on larger contour ensembles. Finally, the interactivity that ID unlocks calls for reimagining contour boxplots for interactive scenarios. For instance, it could be possible to change parameters or weights in the depth function and see them reflected in the contour boxplot in real time.

REFERENCES

[1] R. Serfling and Y. Zuo. “General notions of statistical depth function”. In: *The Annals of Statistics* 28.2 (2000), pp. 461–482. doi: 10.1214/aos/1016218226. url: https://doi.org/10.1214/aos/1016218226.

[2] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi. “A review of uncertainty quantification in deep learning: Techniques, applications and challenges”. In: *Information Fusion* 76 (2021), pp. 243–297. issn: 1566-2535. doi: https://doi.org/10.1016/j.inffus.2021.05.008. url: https://www.sciencedirect.com/science/article/pii/S1566253521001081.

[3] F. Renard, S. Guedria, N. D. Palma, and N. Vuillerme. “Variability and reproducibility in deep learning for medical image segmentation”. In: *Scientific Reports* 10.1 (Aug. 13, 2020), p. 13724. issn: 2045–2322. doi: 10.1038/s41598-020-69920-0. url: https://doi.org/10.1038/s41598-020-69920-0.

[4] M. Zhang, Q. Li, L. Chen, X. Yuan, and J.-H. Yong. “EnConVis: A Unified Framework for Ensemble Contour Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* (2022), pp. 1–1. doi: 10.1109/TVCG.2021.3140153.

[5] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann. “Visual Analysis of Spatial Variability and Global Correlations in Ensembles of Iso-Contours”. In: *Computer Graphics Forum* 35.3 (2016), pp. 221–230. doi: https://doi.org/10.1111/cgf.12898. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12898. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12898.

[6] I. Demir, M. Jarema, and R. Westermann. “Visualizing the Central Tendency of Ensembles of Shapes”. In: *SIGGRAPH ASIA 2016 Symposium on Visualization*. SA ’16. Macau: Association for Computing Machinery, 2016. isbn: 9781450345477. doi: 10.1145/3002151.3002165. url: https://doi.org/10.1145/3002151.3002165.

[7] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. “Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[8] M. Rautenhaus, M. Böttinger, S. Siemen, R. Hoffman, R. M. Kirby, M. Mirzargar, N. Röber, and R. Westermann. “Visualization in Meteorology—A Survey of Techniques and Tools for Data Analysis Tasks”. In: *IEEE Transactions on Visualization and Computer Graphics* 24.12 (2018), pp. 3268–3296. doi: 10.1109/TVCG.2017.2779501.

[9] M. Mirzargar and R. T. Whitaker. “Representative Consensus from Limited-Size Ensembles”. In: *Computer Graphics Forum* 37.3 (2018), pp. 13–22. doi: https://doi.org/10.1111/cgf.13397. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13397. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13397.

[10] P. Voglreiter, P. Mariappan, M. Pollari, R. Flanagan, R. Blanco Sequeiros, R. H. Portugaller, J. Fütterer, D. Schmalstieg, M. Kolesnik, and M. Moche. “RFA Guardian: Comprehensive Simulation of Radiofrequency Ablation Treatment of Liver Tumors”. In: *Scientific Reports* 8.1 (Jan. 15, 2018), p. 787. issn: 2045–2322. doi: 10.1038/s41598-017-18899-2. url: https://doi.org/10.1038/s41598-017-18899-2.

[11] M. Raj, M. Mirzargar, J. S. Preston, R. M. Kirby, and R. T. Whitaker. “Evaluating Shape Alignment via Ensemble Visualization”. In: *IEEE Computer Graphics and Applications* 36.3 (2016), pp. 60–71. doi: 10.1109/MCG.2015.70.

[12] K. Mosler. “Depth Statistics”. In: *Robustness and Complex Data Structures: Festschrift in Honour of Ursula Gather*. Ed. by C. Becker, R. Fried, and S. Kuhnt. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 17–34. isbn: 978-3-642-35494-6. doi: 10.1007/978-3-642-35494-6_2. url: https://doi.org/10.1007/978-3-642-35494-6_2.

[13] S. López-Pintado and J. Romo. “A half-region depth for functional data”. In: *Computational Statistics & Data Analysis* 55.4 (2011), pp. 1679–1695. issn: 0167-9473. doi: https://doi.org/10.1016/j.csda.2010.10.024. url: https://www.sciencedirect.com/science/article/pii/S0167947310004123.

[14] C. Gao. “Robust regression via mutivariate regression depth”. In: *Bernoulli* 26.2 (2020), pp. 1139–1170. doi: 10.3150/19-BEJ1144. url: https://doi.org/10.3150/19-BEJ1144.

[15] R. Jörnsten. “Clustering and classification based on the L1 data depth”. In: *Journal of Multivariate Analysis* 90.1 (2004). Special Issue on Multivariate Methods in Genomic Data Analysis, pp. 67–89. issn: 0047-259X. doi: https://doi.org/10.1016/j.jmva.2004.02.013. url: https://www.sciencedirect.com/science/article/pii/S0047259X04000272.

[16] J. Wang, S. Hazarika, C. Li, and H.-W. Shen. “Visualization and Visual Analysis of Ensemble Data: A Survey”. In: *IEEE Transactions on Visualization and Computer Graphics* 25.9 (2019), pp. 2853–2872. doi: 10.1109/TVCG.2018.2853721.

[17] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead. “Noodles: A Tool for Visualization of Numerical Weather Model Ensemble Uncertainty”. In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1421–1430. doi: 10.1109/TVCG.2010.181.

[18] A. Kumpf, B. Tost, M. Baumgart, M. Riemer, R. Westermann, and M. Rautenhaus. “Visualizing Confidence in Cluster-Based Ensemble Weather Forecast Analyses”. In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), pp. 109–119. doi: 10.1109/TVCG.2017.2745178.

[19] F. Ferstl, K. Bürger, and R. Westermann. “Streamline Variability Plots for Characterizing the Uncertainty in Vector Field Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 767–776. doi: 10.1109/TVCG.2015.2467204.

[20] K. Pothkow and H.-C. Hege. “Positional Uncertainty of Isocontours: Condition Analysis and Probabilistic Measures”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.10 (2011), pp. 1393–1406. doi: 10.1109/TVCG.2010.247.

[21] T. Athawale, E. Sakhaee, and A. Entezari. “Isosurface Visualization of Data with Nonparametric Models for Uncertainty”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 777–786. doi: 10.1109/TVCG.2015.2467958.

[22] K. Pöthkow and H.-C. Hege. “Nonparametric Models for Uncertainty Visualization”. In: *Computer Graphics Forum* 32.3pt2 (2013), pp. 131–140.

[23] S. López-Pintado and J. Romo. “On the Concept of Depth for Functional Data”. In: *Journal of the American Statistical Association* 104.486 (2009), pp. 718–734. doi: 10.1198/jasa.2009.0108. eprint: https://doi.org/10.1198/jasa.2009.0108. url: https://doi.org/10.1198/jasa.2009.0108.

[24] R. Y. Liu. “On a Notion of Data Depth Based on Random Simplices”. In: *The Annals of Statistics* 18.1 (1990), pp. 405–414. issn: 00905364. url: http://www.jstor.org/stable/2241550 (visited on 03/27/2023).

[25] J. W. Tukey. “Mathematics and the Picturing of Data”. In: *Proceedings of the International Congress of Mathematicians, Vancouver, 1975* 2 (1975), pp. 523–531. url: https://cir.nii.ac.jp/crid/1573950399770196096.

[26] Y. Sun and M. G. Genton. “Functional Boxplots”. In: *Journal of Computational and Graphical Statistics* 20.2 (2011), pp. 316–334. doi: 10.1198/jcgs.2011.09224. eprint: https://doi.org/10.1198/jcgs.2011.09224. url: https://doi.org/10.1198/jcgs.2011.09224.

[27] T. Jech. *Set Theory*. 3rd ed. Heidelberg: Springer Berlin, 2003. doi: 10.1007/3-540-44761-X.

[28] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. “Array programming with NumPy”. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. doi: 10.1038/s41586-020-2649-2. url: https://doi.org/10.1038/s41586-020-2649-2.

[29] E. Montagnon, M. Cerny, A. Cadrin-Chênevert, V. Hamilton, T. Derennes, A. Ilinca, F. Vandenbroucke-Menu, S. Turcotte, S. Kadoury, and A. Tang. “Deep learning workflow in radiology: a primer”. In: *Insights into Imaging* 11.1 (Feb. 10, 2020), p. 22. issn: 1869–4101. doi: 10.1186/s13244-019-0832-5. url: https://doi.org/10.1186/s13244-019-0832-5.

[30] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Cham: Springer International Publishing, 2015, pp. 234–241. isbn: 978-3-319-24574-4.

[31] P. F. Raudaschl, P. Zaffino, G. C. Sharp, M. F. Spadea, A. Chen, B. M. Dawant, T. Albrecht, T. Gass, C. Langguth, M. Lüthi, F. Jung, O. Knapp, S. Wesarg, R. Mannion-Haworth, M. Bowes, A. Ashman, G. Guillard, A. Brett, G. Vincent, M. Orbes-Arteaga, D. Cárdenas-Peña, G. Castellanos-Dominguez, N. Aghdasi, Y. Li, A. Berens, K. Moe, B. Hannaford, R. Schubert, and K. D. Fritscher. “Evaluation of segmentation methods on head and neck CT: Auto-segmentation challenge 2015”. In: *Medical Physics* 44.5 (2017), pp. 2020–2036. doi: https://doi.org/10.1002/mp.12197. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.12197. url: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12197.

[32] M. Leutbecher and T. Palmer. “Ensemble forecasting”. In: *Journal of Computational Physics* 227.7 (2008), pp. 3515–3539.

[33] Y. Sun, M. G. Genton, and D. W. Nychka. “Exact fast computation of band depth for large functional datasets: How quickly can one million curves be ranked?” In: *Stat* 1.1 (2012), pp. 68–74.

[34] B. Ma and A. Entezari. “An Interactive Framework for Visualization of Weather Forecast Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), pp. 1091–1101. doi: 10.1109/TVCG.2018.2864815.

5

DEPTH FOR MULTI-MODAL CONTOUR ENSEMBLES

This chapter extends the Inclusion Depth methodology presented in Chapter 4 to handle ensembles exhibiting multi-modal patterns of variation (i.e., containing several representative shapes). Such patterns often arise from differences in training data, modeling ambiguities, or inter-clinician disagreements. We propose the CDclust algorithm and the notion of contour relative depth, which permit detecting and visualizing these modes of variation. Further, we present significant performance improvements for depth computation through a linear-time algorithm and inclusion matrices. This enhancement potentially enables clinicians to disentangle complex ensemble structures, facilitating more targeted quality assessment. This chapter thereby improves upon Inclusion Depth's computational efficiency while expanding its analytical scope, further equipping clinicians with tools to investigate ensemble uncertainty.

This chapter is based on the published paper: Chaves-de-Plaza, Nicolas F., M. Molenaar, P. Mody, M. Staring, R. van Egmond, E. Eisemann, A. Vilanova, and K. Hildebrandt. "Depth for Multi-Modal Contour Ensembles". In: Computer Graphics Forum 43.3 (2024), e15083. doi: 10.1111/CGF.15083

5.1. INTRODUCTION

The problem of analyzing the distributional properties of contour ensembles arises in a wide range of domains like meteorology, where analysts need to interpret multiple simulation runs [1]; medicine, where clinicians plan interventions using robust representations of the organs [2]; and biology [3], where changes in cells' morphology across a population of cells can be indicative of looming disease. The contour depth methodology has become established to visually analyze contour ensembles in terms of their representatives, confidence bands, and outliers. Examples include analyzing variations of meteorological forecasts [4] and determining representative and outlying contours in medical image segmentations [5, 6].

There are two contour depth notions available: Inclusion Depth (ID) [6] and Contour Band Depth (CBD) [4]. ID assesses the number of times the contour contains and is contained by other contours. CBD determines the centrality of a contour by counting the number of times it falls in the band formed by tuples of other contours in the ensemble. When dealing with real data, contours tend to intersect multiple times. Non-nested pairs of contours do not contribute to the depth score, resulting in less discriminative CBD and ID depths. To overcome this challenge, epsilon ID (eID) and eCBD consider partial containment. The depth scores that ID and CBD yield can be used to summarize contour ensembles in terms of their representatives, confidence bands, and outliers, which can be visualized using contour boxplots [4].

The main practical limitation of contour depth methods is their scalability. Most practical implementations of CBD only consider bands formed by pairs of contours. Even then, given that there are N^2 bands formed by pairs of contours in a N-contour ensemble, CBD takes $\mathcal{O}(MN^3)$ operations to compute an ensemble's depth, where M is the resolution of the domain used to perform the contour comparisons. By only considering pairwise relationships, ID provided an order-of-magnitude speedup taking $O(MN^2)$ time, without sacrificing performance (i.e., ID and CBD yield comparable depth estimates). Nevertheless, this might not be sufficient in use cases that require multiple depth evaluations like interactive analysis of large contour ensembles and clustering [7].

In this paper, we accelerate the computation of ID. In particular, we present a linear time algorithm for computing eID that leverages precomputed inclusion fields. Computing a contour's depth reduces to querying these fields in $\mathcal{O}(M)$ time. Moreover, we introduce the inclusion matrix, which encodes the inclusion relationship between pairs of contours, for accelerating the recomputation of an ensemble's depths when adding or removing groups of contours, without requiring to recompute the whole ensemble's depths. The ability to quickly recompute depths is useful when computing depths progressively [8] or when updating an ensemble's configuration based on user interaction;

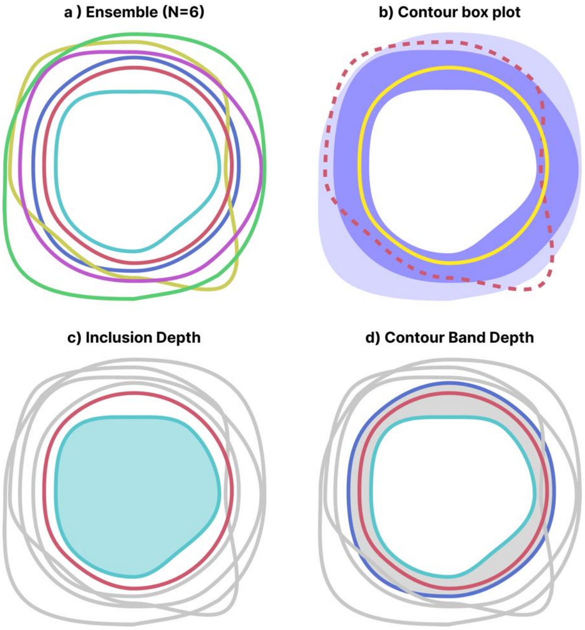


Figure 5.1.: Computation of Inclusion Depth (ID) and Contour Band Depth (CBD) for the six-member ensemble in (a). In (c), ID involves evaluating containment relationships between contour pairs. CBD (d) counts the number of times contours fall within bands defined by a subset of contours, shown in purple and blue. Additionally, (b) presents depth scores through contour boxplots, providing a statistical summary of the ensemble with median (yellow), confidence bands (light and dark purple), and outliers (dashed red line).

and critical when using procedures that require multiple calls to the depth function like clustering.

A limiting assumption of existing contour depth methods is that contours in the ensemble were drawn from the same distribution. In practical scenarios with multiple modes of variation, global depth analysis may produce unexpected results, such as assigning high-depth scores to points that are outliers within one mode but centrally located in the overall ensemble [9].

We overcome the uni-modality assumption by introducing an extension of the contour depth framework for multi-modal ensembles. Central to this extension is the use of relative depth (ReD). By optimizing the ensemble's average ReD, the CDclust algorithm disentangles its modes of variation. Each iteration of CDclust entails calling a contour depth procedure several times on subsets of the data. Therefore, crucial to CDclust's practical application are the newly introduced fast depth computation schemes. Through experiments with synthetic datasets, we illustrate how ReD and CDclust facilitate non-parametric analysis of multi-modal ensembles. Additionally, we show two case studies in the fields of medical image segmentation and meteorological forecasting that further demonstrate the practical utility of the multi-modal depth toolkit.

In summary, our main contributions are:

- Schemes for accelerated computation and recomputation of contour depths, in particular, a linear time algorithm for eID and the inclusion matrix, which removes the dependency on the contours' domain resolution when recomputing depths on subsets of the ensemble. These speedups are crucial to enable use cases like progressive depth computation and clustering.
- The first framework for multi-modal depth analysis of contour ensembles. The CDclust algorithm leverages the inclusion matrix to disentangle modes of variation in a contour ensemble by maximizing its average relative depth.

5.2. RELATED WORK

Our research advances uncertainty visualization methods when using ensembles to characterize underlying distributions. Ensembles permit quantifying uncertainty related to initial conditions, training data, or model parameters [10]. When visualizing ensembles, the data type, dimensionality, and analytical tasks must be considered [11]. We focus on ensembles of contours derived from spatial data, addressing scenarios like thresholding scalar fields.

Spaghetti plots are commonly used to display contour ensembles, but they become cluttered and less trustworthy for larger ensembles [12, 13]. Our focus is on providing an overview of the statistical properties of the ensemble such as its representatives, confidence bands, and outliers. Existing methods are categorized into parametric and non-parametric approaches. Parametric methods assume a distribution, such as Gaussian models fitted to contours' PCA-reduced signed distance fields (SDF) [14, 15] or Gaussian models at each grid point [16]. Non-parametric methods, like Contour Probability Plots [17] and EnConVis [18], avoid distributional assumptions and offer accurate point-wise descriptions. A hybrid approach uses pairwise contour comparisons to determine centrality [19].

Contour depths, a nonparametric method, exhibit desirable properties such as sensitivity to shape and topology, making them suitable for downstream analyses like clustering [7] and regression [20]. Contour Band Depth [4], while effective for ensemble characterization, scales poorly with the size of the ensemble. A recently proposed alternative with more favorable scaling behavior is the Inclusion Depth [6]. In this paper, we unify both depth notions using the inclusion matrix, capturing the topological relationships among ensemble members. The proposed approach achieves an order of magnitude speedup in Contour Band Depth (CBD) and accelerates the recomputation of depths, which is relevant in interactive scenarios and clustering. Furthermore, we present a linear algorithm for epsilon Inclusion Depth (eID), enabling using eID with large contour ensembles.

Depth methods, assuming a uni-modal distribution, may yield unexpected results in the presence of multiple modes. Previous research addresses mode variation in contour ensembles through clustering. We leverage depth to support this process. Notable approaches include detecting multi-scale symmetries using high-dimensional transform-invariant spaces and nearest neighbor search [21]; and using lower-dimensional representations like PCA-reduced contours SDFs [14] with existing clustering methods such as KMeans [17], density-based clustering [22] and agglomerative hierarchical clustering [15], which favors compact elliptical clusters for Gaussian mixture model fitting [14]. Finally, the EnConVis framework for contour ensemble analysis emphasizes the importance of the distance function in clustering and classification tasks [18].

Depth methods enhance clustering but are yet to be explored in contour contexts. Notable instances include a scheme for clustering multi-variate data using l1 depth [7], recently adapted to use curve depth [23]; the bisecting k-spatialMedian algorithm based on spatial or l1 depth [24]; depth-based clustering analysis (DBCA) for affine-invariant and noise-robust clustering [25]; CRAD, a density-based clustering algorithm employing robust data depth [26]; the depth difference (DeD) metric for determining optimal cluster count [27], and depth-based medoids clustering algorithm (DBMCA) for high-dimensional directional data [9].

5.3. BACKGROUND: CONTOUR DEPTH

The contour statistical depth methodology permits characterizing an ensemble of contours in terms of the centrality, or alternatively outlyingness, of their members. In the following, we discuss the two main notions of contour depth: Inclusion Depth and Contour Band Depth. Figure 5.1 illustrates the available contour depth notions and how to visualize an ensemble's summary statistics using contour boxplots.

Inclusion Depth Let C be an ensemble of N contours. The Inclusion Depth (ID) of $c_i \in C$ results from the number of other contours that c_i

contains and in which c_i is contained [6]:

$$\begin{aligned} \mathrm{ID}(c_i|C) &= \frac{2}{N}\min\{IN_{in}(c_i), IN_{out}(c_i)\} \text{ with} \\ \mathrm{IN}_{in}(c_i) &= \sum_{j=1}^{N} in(c_i) \subset in(c_j), \text{ and} \\ \mathrm{IN}_{out}(c_i) &= \sum_{j=1}^{N} in(c_j) \subset in(c_i), \end{aligned} \tag{5.1}$$

where $in(c_i)$ denotes the subset in the plane enclosed by the contour and $\subset$ yields 0 or 1, depending on the contours' inclusion relationship. The ID values range between $[0,1]$. When using bitmaps of M pixels to represent contours, ID has a computational complexity of $\mathcal{O}(MN^2)$.

Contour Band Depth The Contour Band Depth (CBD) of $c_i \in C$ is the average number of times that the contour falls inside the band formed by any other J-band with $J \in \{2,3,4,\dots,N-1\}$ [4]. We say a contour c_i falls in the band formed by J other contours if it contains the contours' intersection and is contained by their union:

$$CB(c_i|c_1,\dots c_j) = \bigcap_{j=1}^{j} in(c_j) \subset in(c_i) \textbf{ and } in(c_i) \subset \bigcup_{j=1}^{j} in(c_j) \tag{5.2}$$

Contour Band Depth (CBD) can be written as

$$\mathrm{CBD}(c_i|C) = \sum_{j=2}^{J} \frac{1}{\binom{N}{j}} \sum_{k=1}^{\binom{N}{j}} CB(c_i|B_k^j), \tag{5.3}$$

where B_k^j is the k^{th} band of the set of j-contours bands. The CBD values range between [0, 1]. CBD is computationally expensive for $J > 2$, so, in practice, $J = 2$ is used. In Sec. 5.5, we illustrate how to obtain compute CBD in $\mathcal{O}(N^2)$ time using the inclusion matrix.

Epsilon Contour Depth When contours intersect, there tend to be ties (*i.e.*, pairs of contours for which neither contains the other) and low depth scores. To mitigate this, variants of CBD and ID have been introduced that use the modified epsilon subset operator

$$A \subset_\epsilon B = 1 - \begin{cases} 0 & |A| = 0, \\ |A-B|/|A| & \text{otherwise,} \end{cases} \tag{5.4}$$

where $|A|$ denotes the area of A, $A-B$ the relative set difference and $\subset_\epsilon$ outputs a continuous value between $[0,1]$.

The modified epsilon ID (eID) [6], replaces $\subset$ in Eq. 5.1 with $\subset_\epsilon$. Similarly, the modified CBD, which we will refer to as epsilon CBD (eCBD), replaces $\subset$ in Eq. 5.2 with $\subset_\epsilon$, yielding the epsilon band containment operator

$$CB_\epsilon(c_i|c_1,\ldots c_j) = \min\left(\bigcap_{j=1}^{j} in(c_j) \subset_\epsilon in(c_i), in(c_i) \subset_\epsilon \bigcup_{j=1}^{j} in(c_j)\right), \qquad (5.5)$$

Computing eID takes $\mathcal{O}(MN^2)$ time.

Computing eCBD entails forming a $N \times \sum_J \binom{N}{J}$ matrix listing the outputs of Eq. 5.5. Individual depth values are then computed by thresholding and averaging matrix entries. Because eCBD requires assembling the complete matrix, it is not possible to apply the same acceleration strategy as for CBD. Therefore, eCBD has a complexity of $\mathcal{O}(MN\sum_J \binom{N}{J})$.

5.4. LINEAR EPSILON INCLUSION DEPTH COMPUTATION

The Epsilon Inclusion Depth (eID) replaces the subset operator in Eq. 5.1 by the epsilon subset operator, defined in Eq. 5.4, to compute the proportion of area of one contour that is contained in another ($\text{IN}_{in}^{\epsilon}$ and $\text{IN}_{out}^{\epsilon}$). By reorganizing the loops in these expressions, it is possible to obtain an algorithm to compute eID in $\mathcal{O}(NM)$. In the following, we simplify notation by using $c_i = in(c_i)$. $c_i(m)$ yields c_i's value at the m^{th} domain point.

Eq. 5.6 provides the derivation for $\text{IN}_{in}^{\epsilon}$. We start by plugging Eq. 5.4 into IN_{in} in Eq. 5.1. Note that the set difference can be written as a loop over the M bitmap pixels of a contour, where $c_i(m) = 1$ if pixel m is in contour i, and 0 otherwise. We compute $\sum_{j=1}^{N}(1-c_j(m))$ ahead of time and store it in a lookup table $\text{pre}_{in}^{\epsilon}(m) = \sum_{j=1}^{N}(1-c_j(m))$. Computing these values takes $\mathcal{O}(MN)$ time but only needs to be done once for all contours.

$$
\begin{aligned}
\mathrm{IN}^{\epsilon}_{in}(c_i) &= \sum_{j=1}^{N} 1 - \frac{|c_i - c_j|}{|c_i|} \\
&= N - \frac{1}{|c_i|}\sum_{j=1}^{N} |c_i - c_j| \\
&= N - \frac{1}{|c_i|}\sum_{j=1}^{N}\sum_{m}^{M} (1 - c_j(m))c_i(m) \\
&= N - \frac{1}{|c_i|}\sum_{m}^{M} c_i(m)\sum_{j=1}^{N}(1 - c_j(m)) \\
&= N - \frac{1}{|c_i|}\sum_{m}^{M} c_i(m)\mathrm{pre}^{\epsilon}_{in}(m)
\end{aligned}
\tag{5.6}
$$

The same idea also applies to $\mathrm{IN}^{\epsilon}_{out}$. We again refactor the formula to obtain a precomputed lookup table $\mathrm{pre}^{\epsilon}_{out}(m) = \sum_{j=1}^{N} \frac{c_j(m)}{|c_j|}$ which is shared between all contours. Computing $\mathrm{IN}^{\epsilon}_{out}(c_i)$ and $\mathrm{IN}^{\epsilon}_{in}(c_i)$ now takes $\mathcal{O}(M)$ time with a precomputation of $\mathcal{O}(MN)$ to create the lookup tables. This results in $\mathcal{O}(MN)$ time complexity to compute eID for all N contours in the ensemble.

$$
\begin{aligned}
\mathrm{IN}^{\epsilon}_{out}(c_i) &= \sum_{j=1}^{N} 1 - \frac{|c_j - c_i|}{|c_j|} \\
&= N - \sum_{j=1}^{N} \frac{|c_i - c_j|}{|c_j|} \\
&= N - \sum_{j=1}^{N}\sum_{m}^{M} \frac{(1 - c_i(m))c_j(m)}{|c_j|} \\
&= N - \sum_{m}^{M}(1 - c_i(m))\sum_{j=1}^{N}\frac{c_j(m)}{|c_j|} \\
&= N - \sum_{m}^{M}(1 - c_i(m))\mathrm{pre}^{\epsilon}_{out}(m)
\end{aligned}
\tag{5.7}
$$

5.5. FAST DEPTH RECOMPUTATION

In the following, we introduce the inclusion matrix, which permits decoupling the depth computation from the assessment of the pairwise inclusion relationship between contours. We show how, in practice, this

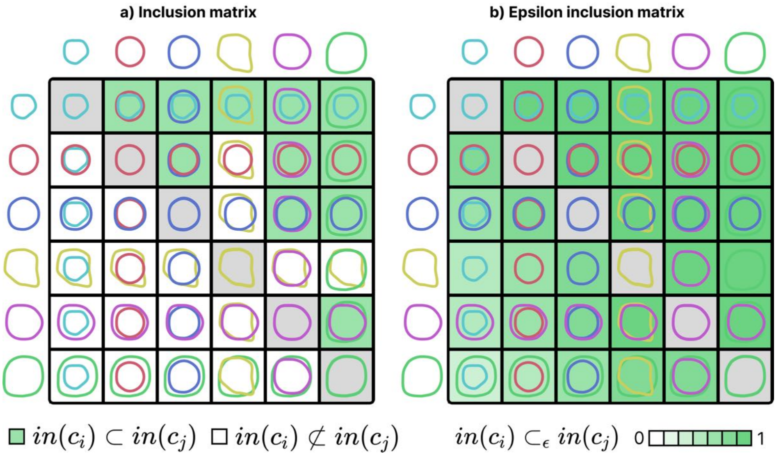

a) Inclusion matrix
b) Epsilon inclusion matrix
🟩 $in(c_i) \subset in(c_j)$ □ $in(c_i) \not\subset in(c_j)$
$in(c_i) \subset_\epsilon in(c_j)$ 0 1

Figure 5.2.: Inclusion (a) and epsilon inclusion (b) matrices of the contour ensemble in Fig. 5.1. In the strict inclusion matrix, cells are colored if a row contour is a subset of the column contour. The epsilon inclusion matrix values range between 0 and 1, discretized into seven bins for visualization simplicity.

translates to a significant speedup in the computation of ID and CBD ($J = 2$) on an ensemble's subsets, a feature critical for use cases that require depth evaluations within the ensemble, like clustering.

At the hearts of ID and CBD are the subset and epsilon subset operators, which permits establishing the containment relationship between all pairs of contours in the ensemble. We term the matrix that collects all the pairwise comparisons inclusion matrix $\mathfrak{C}$ and epsilon inclusion matrix $e\mathfrak{C}$, respectively. Starting with the latter, a cell $e\mathfrak{C}_{ij}$ with $i, j \in N$ is computed as:

$$e\mathfrak{C}_{ij} = in(c_i) \subset_\epsilon in(c_j) \tag{5.8}$$

where $\subset_\epsilon$ is the operator defined in Eq. 5.4. To obtain $\mathfrak{C}$, it suffices to threshold $e\mathfrak{C}$ as

$$\mathfrak{C}_{ij} = \mathbb{1}_{\geq 1}[e\mathfrak{C}_{ij}], \tag{5.9}$$

where $\mathbb{1}[\cdot]$ is the indicator function.

Fig. 5.2 depicts $\mathfrak{C}$ and $e\mathfrak{C}$ for an ensemble of six contours. The epsilon inclusion matrix (b) has values that range between 0 and 1, with one denoting full containment. In practice, entries are only zero if the two contours are disconnected components. If this is not the case and $A \not\subset_\epsilon B$ in Eq. 5.4, then the entry will be lower than one but not zero, systematically increasing the depth scores, but preventing ties due to the non-perfect nestedness of contours. In general, the inclusion matrices are not symmetric. For example, $\mathfrak{C}$ is not symmetric as for $i \neq j$, if $in(c_i) \subset in(c_j)$ then $in(c_j) \not\subset in(c_i)$. It is also not antisymmetric because $in(c_i), in(c_j)$ might not share a containment relationship like in the case where they are disconnected components.

The inclusion matrix provides the information needed to compute CBD when only bands formed by two contours are considered. Therefore, in the particular case of CBD with $J = 2$, it is possible to obtain a quadratic runtime. It is possible to determine the number of bands a function falls in by calculating the number of functions above (N_a) and below (N_b) that function, and using the formula $N_{bands} = N_a N_b + N - 1$ [8]. This simplification works because of the assumption that a function cannot fall in a band formed by functions that cross over [1]. In the contour case, by setting $N_a = \mathrm{IN}_{out}$ and $N_b = \mathrm{IN}_{in}$, both of which can be obtained from the inclusion matrix, it is possible to obtain CBD in $\mathcal{O}(MN^2)$, the time it takes to compute the inclusion matrix. It must be noted that this strategy does not apply to eCBD because eCBD requires operating on the full contours-vs-bands matrix.

The inclusion matrix decouples the initial computation of the pairwise inclusion relationships from the depth calculations. Therefore, adding or removing small subsets of contours is fast. Adding N' new contours to the ensemble grows the inclusion matrix from N^2 to $(N + N')^2$ entries. Adding these $2NN' + N'^2$ new entries takes $\mathcal{O}(MNN' + MN'^2)$ time, significantly faster than recomputing the matrix from scratch. In the next section, we will show how this feature enables CDclust. Additionally, in the experiments section, we show how it can be used to progressively compute depth.

5.6. MULTI-MODAL ANALYSIS

5.6.1. RELATIVE DEPTH

Relative Depth (ReD) is an extension of the concept of depth to multiple clusters or modes of variation. Intuitively, a contour belongs to the correct partition if the contour's depth in the partition it belongs to is higher than what it could attain if it belonged to any other partition. In the following, we refer to the former as depth-within and to the latter as depth-between.

Let I_K be a partitioning of the N contours into K clusters. $I_K(k)$ yields the ids of the contours belonging to partition k. Given a contour $c_i \in C$ with $i \in I_K(k)$, we compute its relative depth ReD_i as

$$\mathrm{ReD}_i = \mathrm{ReD}(c_i | C, I_K) = D_i^w - D_i^b \qquad (5.10)$$

with the depth-within defined as

$$D_i^w = D(c_i | \{c_j | j \in I_K(k)\}), \qquad (5.11)$$

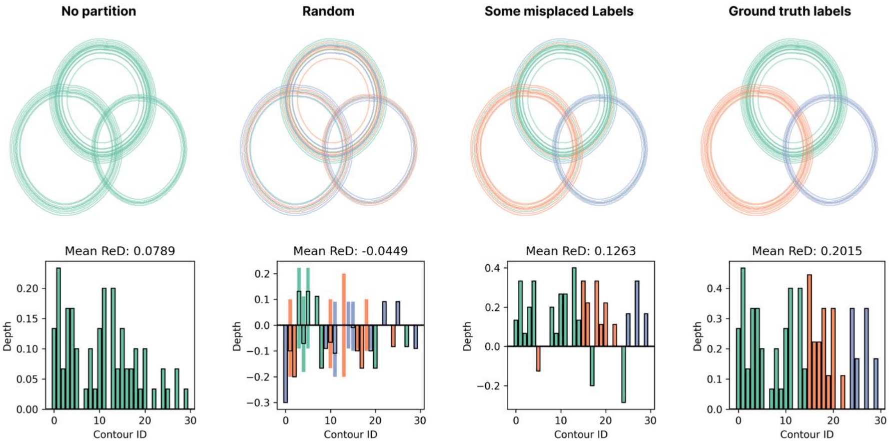


Figure 5.3.: Relative depth scores as a function of the clustering labels for an ensemble of $N = 30$ contours in a three-ring configuration. Each ring has a different proportion of contours. The top row illustrates the different label assignments. The bottom row depicts the depth-within cluster (bar above 0 line), depth-between cluster (bar below 0 line), and relative depth (bar with black stroke and no fill) for each ensemble member.

and depth-between as

$$D_i^b = \max_{l \neq k; l \in \{1,\ldots,K\}} D(c_i | I_K(l)), \tag{5.12}$$

where D is any suitable contour depth notion like Inclusion Depth or Contour Band Depth. ReD values range between [-1, 1]. A contour that attains the minimum value in this range is likely assigned to the wrong partition or corresponds to an outlier because its D^w is zero and its D^b is the maximum value. In contrast, a contour with the maximum value of the range is considered the median of the partition it belongs to.

Fig. 5.3 depicts the ReD (using ID) per contour for different partitionings of an ($N = 30$) ensemble of contours made of overlapping rings spawned in different locations with perturbed radius. The first row shows the ensemble and its partitioning with each partition colored differently. The second row depicts the D_i^w (colored bar above zero line), D_i^b (mirrored colored bar below zero line), and ReD_i (non-colored bar with black stroke) per contour (horizontal axis). The first column represents the unimodal case in which calculating the ReD reduces to computing the depth-within of each ensemble member. The other three columns show a random partitioning, a partitioning in which only some labels were exchanged, and the generative ground truth labels. It can be observed how the average ReD is maximized by the partitioning with generative labels because there are no contours with non-zero D_i^b.

Interestingly, the average ReD in the case with ground truth labels is also larger than in the uni-modal case, despite the latter not having contours with positive depth-between. This shows how the incorrect uni-modal assumption of the traditional depth notion negatively affects overall depth scores. In the experiments, we leverage this observation to show how ReD can be used as a cluster validation tool to determine the optimal number of clusters K.

5.6.2. CDCLUST

The average ReD score of a partitioning I_K provides an indication of its quality. Specifically, we say that I_K is satisfactory if the average ReD is maximized, which entails maximizing the depth-within and minimizing the depth-between of every contour. The problem of obtaining the I_K that maximizes ReD can be formulated as

$$I_K = \underset{I_K}{\operatorname{argmax}} \frac{1}{N} \sum_{i=1}^{N} \text{ReD}(c_i | C, I_K) = \frac{1}{N} \sum_{i=1}^{N} D_i^w - D_i^b, \tag{5.13}$$

where C is fixed.

The optimization problem in Eq. 5.13 has a large discrete search space. We adopt a heuristic inspired by KMeans [28] to obtain a reasonable solution. Algorithm 2 presents the pseudocode of CDClust. CDclust takes as input the contour ensemble C, the desired number of components K, random trials T, and iterations it_{max}. In practice, there are potentially many local optima. Additionally, in some cases, a cluster might become empty. To ensure a better exploration of the solution space, we permit the user to define a number of random trials to perform.

Starting from a random partitioning, CDclust proceeds to iteratively increase the partitioning depth by reassigning contours to the cluster that represents them best. Specifically, at each iteration, the algorithm computes the contours' depth with respect to the other clusters and collects these depth values in the matrix $D_K \in \mathbb{R}^{N \times K}$. We define the competing cluster of a contour as the cluster that maximizes its depth

$$I_{comp} = \underset{l \in \{1,\ldots,K\}}{\operatorname{argmax}} D(c_i | I_K(l)). \tag{5.14}$$

If the current assignment $I_K(c_i)$ maximizes the contours' depth, then it is not relocated. Otherwise, the algorithm reassigns to its competing cluster.

Algorithm 2 Depth-Based Contour Clustering (CDclust)

```
Require: $C, K, T, it_{max}$    ▷ $N$-contour ensemble, number of components, number of random trials and of iterations
$I_K^* \leftarrow \{\}$    ▷ Best partition
$\mu ReD^* \leftarrow -\infty$    ▷ Best average ReD
for $t \in \{1, \dots, T\}$ do
    $I_K \leftarrow$ random partitioning of $C$ into $K$ clusters
    for $i \in \{1, \dots, it_{max}\}$ do
        $D^K \in \mathbb{R}^{N \times K}$    ▷ Between-cluster depth matrix
        for $k \in 1, \dots, K$ do
            $D^K_{\cdot,k} \leftarrow \{D(c_i|C_k)|c_i \in C\}$    ▷ Via inclusion matrix
        end for
        $D^w \leftarrow \{D^K_{i,k}|k = I_K(c_i)$ and $i = \{1, \dots, N\}\}$
        $D^b \leftarrow \{D^K_{i,l_i}|l_i = \text{argmax}_{l_i \neq I_K(c_i)} D^K_{i,l_i}$ and $i = \{1, \dots, N\}\}$
        $I'_K \leftarrow I_K$
        $I_K \leftarrow \{\text{argmax}_k D^K_{i,k}|i \in \{1, \dots, N\}\}$
        $\mu ReD \leftarrow \frac{1}{N} \sum_i D^w_i - D^b_i$
        if $\mu ReD > \mu ReD^*$ then
            $I^{K*} \leftarrow I_K$
            $ReD^* \leftarrow ReD$
        end if
        if $I_K = I'_K$ then
            return $I_K$
        end if
    end for
end for
return $I_K$
```

5.6.3. CDCLUST COMPLEXITY

CDclust's runtime depends on the number of trials T and a maximum number of iterations it_{max}. Within each iteration, CDclust requires computing the depth of each contour with respect to each cluster. If the inclusion matrix is used, then its precomputation is the bottleneck of the algorithm taking $\mathcal{O}(MN^2)$ time. Within the loop, it takes $\mathcal{O}(N)$ time per contour to compute its depth with respect to all clusters, yielding a complexity of $\mathcal{O}(N^2)$. Therefore, in this case, CDclusts complexity is $\mathcal{O}(MN^2 + it_{max}TN^2)$.

When using the linear time eID, one needs to compute the inclusion fields at each iteration, which takes $\mathcal{O}(MN)$ time. The most expensive part of the algorithm is the computation of the between-cluster depth matrix, which takes $\mathcal{O}(KN)$ time. In total, CDclust with linear eID runs in $\mathcal{O}(it_{max}TMN + it_{max}TKN)$. Note that when a high resolution grid in the plane is used to resolve the contours, MN may be larger than N^2. In this case, CDclust with the linear time eID has slower iterations than CDclust with the inclusion matrix.

5.7. EXPERIMENTS ON SYNTHETIC DATA

This section presents the results of experiments with synthetic datasets, demonstrating the performance of the proposed methods. The experimental code[1] and contour-depth Python package [2] are available as GitHub repositories. Further speedups can be achieved by using a more performant programming language and implementing parallelism in the code. We ran all the experiments on a Mac Book Pro (2022) with an M1 Pro processor (without GPU acceleration) and 32 GB RAM.

5.7.1. FAST COMPUTATION OF CONTOUR DEPTH

Setup We use the shape outside outliers detailed in the Inclusion Depth paper [6]. We define a stochastic model from which we can sample inlier shapes and outlier shapes with higher amplitude and frequency, endowing them with distinct shapes. We use an outlier contamination proportion of 0.1. The second column of Fig. 5.7 shows an example of the shape outlier dataset.

To generate datasets of varying sizes, we start with the full ensemble ($N = 300$) and sample increasingly smaller -nested- subsets in increments of 10 until 10 elements remain, yielding sampling sizes $Ns = \{10, 20, 30, \dots, 300\}$. For the unoptimized CBD in the scaling behavior experiment, we only consider until $N = 150$ due to its steep increase of computational cost. For each combination of method/sample size, we run five random trials to derive confidence intervals of the

[1] https://graphics.tudelft.nl/paper-multimodal-contour-depth
[2] https://graphics.tudelft.nl/contour-depth

results. Finally, for the progressive depth calculation experiment, we use $N = 150$. To increase difficulty, we shuffle the shapes in the ensemble, interleaving inliers and outliers.

Scaling Behavior Fig. 5.4 compares the runtimes of the linear eID computation with other contour depth methods. In particular, the figure includes strict CBD ($J = 2$) and ID. We differentiate whether the method was optimized or not. Optimized CBD refers to computing strict CBD using the expression presented in Sec. 5.5, ID has no optimized version and unoptimized eID refers to using the inclusion matrix to compute the depths. The performance gains are evident. ID and unoptimized eID are at least an order of magnitude faster than CBD when more than two contours are used to form the band. Linear eID is, in turn, an order of magnitude faster than methods based on the inclusion matrix, computing depths of 300-contour ensembles in under ten seconds. These results confirm the speed-ups that ID and linear eID achieve. It is important to note that speed is only one factor to consider when selecting a depth notion. In practice, the properties of strict depth notions might be desired. In this case, the best-performing methods, optimized CBD ($J = 2$) and ID, have a time complexity of $\mathcal{O}(MN^2)$. In the case of CBD, if more bands are desired, the performance of the methods will rapidly degrade as it depends on the number of possible bands that can be formed out of J contours.

Progressive Depth Computation We now demonstrate the usage of fast depth computation for progressively calculating and rendering depths, which can enhance analytical processes [29]. Fig. 5.5 compares the runtimes of the batched and progressive depth computation of a $N = 100$ ensemble. We assume that the ensemble's contours become available one at a time. For the batched method, we recompute the ensemble's ID every time a new contour arrives. For the progressive method, we only compute missing entries of the inclusion matrix and then perform a depth update of the ensemble. As can be observed in the line plot, the cost of adding a contour to the ensemble is significantly higher for the batched version. The $N = 100$ ensemble takes an average of 57 seconds per contour with the first one taking a fraction of a second and the last one more than four minutes. In contrast, the progressive version takes advantage of the information contained in the inclusion matrix to avoid unnecessary recomputations. It takes 1.15 seconds on average to recompute the ensemble's depths, which means that the whole ensemble can be progressively rendered in less than two minutes, allowing for interactive rates. The vertical stripe on the right side of the figure illustrates how the incremental calculation of depth works.

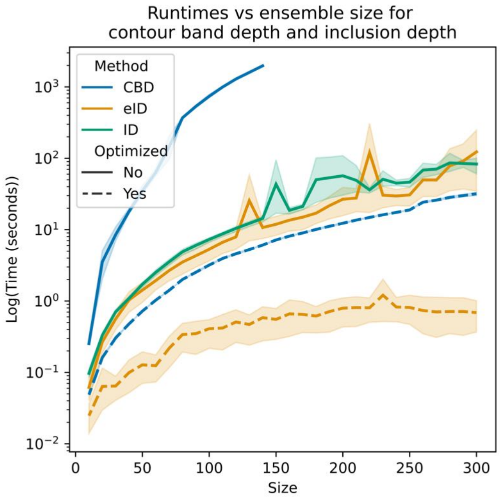


Figure 5.4.: Comparison of mean runtimes for different sample sizes of optimized and unoptimized versions of CBD, ID, and eID. The y-axis uses a logarithmic scale and the shaded area denotes the 95 percent confidence interval across replications.

5.7.2. MULTI-MODAL CONTOUR ANALYSIS

Setup We use three datasets ($N = 100$) that contain multiple modes of variation. First, the three rings dataset has three overlapping groups of circles each with perturbed radii and centers. Each circle group has a different number of circles and spread (different radii distribution). Second, the non-nested cluster dataset contains three groups of circles C1, C2, and C3 arranged such that C1 and C2, and C1 and C3 are nested but C2 and C3 are not. Circles in each group have perturbed radii and centers and different spreads. Finally, we reuse the shape outlier dataset from the last subsection, which can be thought of as an ensemble with two modes of variation: inliers and outliers. Figs. 5.6 and 5.7 illustrate these datasets.

In preliminary experiments, we observed that the performance of CDclust decreases when using ID due to the method's tendency to yield ties if the contours intersect. Therefore, unless mentioned otherwise, we use eID as a depth notion for both ReD and CDclust. For CDclust, we use $T = 5$ and $it_{max} = 10$. The number of clusters K changes depending on the experiment's purpose.

We compare CDclust against two relevant existing methods that leverage a PCA-reduced SDF representation of the contours. To obtain a contour's SDF representation, we compute the signed distance of each

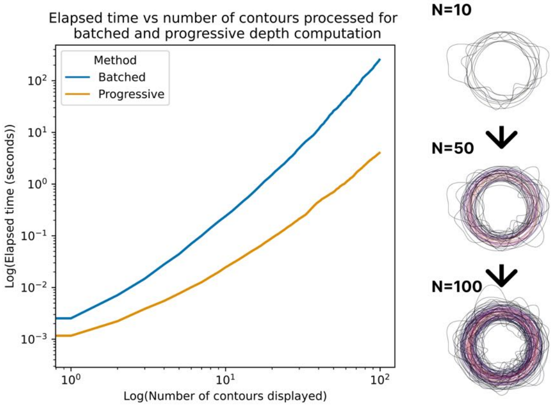


Figure 5.5.: Comparison of the time it takes to compute depths of a growing ensemble ($N = 100$) using batched and progressive depth calculation. The x and y-axes have log scales. The x-axis indicates how many contours have been processed at the time given by the corresponding point in the y-axis. The strip to the right depicts the updating of the depth scores as the ensemble grows.

pixel to the closest point on the contour and use principal components analysis to keep the dimensions in the resulting field that explain 0.999 of the variance [14]. First, we consider KMeans [28], which iteratively improves the clustering by assigning points to the closest center. Similarly to CDclust, we set the number of attempts to 5 and the maximum number of iterations to 10. Second, we consider agglomerative hierarchical clustering (AHC) with average linking, which is part of the CVP pipeline proposed in [14, 15]. We choose the number of clusters to match the one used in CDclust and KMeans. For both clustering algorithms, we use Sklearn's implementation with Euclidean distance as the distance metric.

Cluster Validation Using ReD Average ReD (μReD) can be used to determine the optimal number of clusters. Fig. 5.6 depicts this cluster validation strategy for the three rings dataset ($N = 100$). For different values of K, we run CDclust and compute the clustering μReD. To reduce the sensitivity to a specific clustering result, we perform this process ten times, varying CDclust's random seed. The graph (top row of Fig. 5.6) shows the mean μReD per K surrounded by a 95% confidence interval. It can be observed how $K = 3$, the desired clustering, consistently maximizes μReD. As K increases, the mean μReD decreases and the uncertainty in the clustering results increases. The figure's bottom section shows the resulting clusterings for one of the random seeds. As can be observed, in some cases higher K clusterings preserve the inlier structure of $K = 3$, assigning magnitude outliers to the extra clusters. The depth-within of the swapped contours does not change because of their outlier status, but their depth-between increases, which results only in a slight decrease in μReD. In other cases, a ring group is split into two or more components, reminiscing clusterings obtained with hierarchical methods.

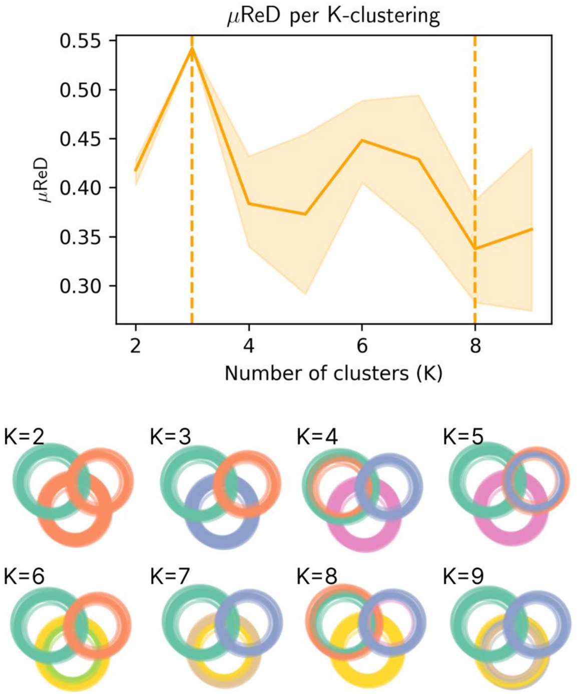


Figure 5.6.: Selection of the optimal number of clusters using μReD. The line plot depicts the mean μReD per K-clustering across ten samples. The shaded area corresponds to a 95% confidence interval. $K = 3$ and $K = 8$ (vertical dashed lines) attain the highest and lowest mean μReD, respectively. The bottom section presents examples of the resulting clusterings for one of the samples.

Comparative Evaluation of CDclust In the synthetic datasets we considered, we observed that ReD, KMeans, and AHC exhibited a similar clustering behavior when using the ground truth K. The methods' behavior changed when exploring alternative Ks. The first column of

Fig 5.7 presents an example of the non-nested cluster dataset. We clustered the dataset with $K = 2$. Both CDclust and AHC put the small group of contours (in orange for CDclust and AHC) in a separate cluster. In contrast, KMeans classified these contours as belonging to the same cluster as the inner ones, which are partially disconnected/unnested. This example shows how CDclust has increased sensitivity to the nestedness relationship between contours, which could be useful in cases where one wants to flag groups of contours with a different nestedness relationship.

The previous result hints at the strength of contour depth in identifying shape outliers. For ID, the user must select a depth threshold for the outliers. CBD uses an automatic mechanism to determine it. We explored the utilization of clustering methods to identify the shape outliers, using the shape outliers dataset and $K = 2$. The second column of Fig. 5.7 shows an example of the results. As can be observed, CDclust assigned all the 16 shape outliers to the same group. It also assigned contours with extreme magnitudes to the group of outliers, highlighting as inliers the highly central core of circular contours. KMeans and AHC, relying on distances rather than on the inclusion relationships, do not achieve a clear separation. KMeans splits the shape outliers, assigning 11 to the green cluster and 5 to the orange one. Furthermore, in the orange cluster KMeans mixes representative contours with shape outliers. For K=2, AHC only separated one magnitude outlier from the rest, combining shape outliers and inliers. This result is sensitive to AHC's linking method and the K used. When we tried larger values for K, AHC assigned shape outliers to low cardinality clusters, producing a satisfactory separation of the inliers similar to CDclust's. Nonetheless, needing to tinker with both K and the linking method hinders the method's practicality. The results indicate that CDclust can also be used to separate outliers from inliers, permitting us to automatically obtain a robust outlier-free cluster, which can be used in downstream procedures.

5.8. CASE STUDIES

In this section, we demonstrate how CDclust+ReD can be used to perform non-parametric multi-modal visual analysis of real datasets. We use eID because it is the fastest depth notion available and because contours in real data tend to intersect. For CDclust, we use the same configuration as in the previous section. For CVP, we implement the pipeline as described in [14]. In summary, CVP uses agglomerative hierarchical clustering of the PCA-reduced SDF representations of the contours to find the modes of variation. The cluster representatives are the geometric medians of each cluster in PCA space. The bands are computed from the SDFs by adding and subtracting from the mean SDF a user-selected number of standard deviations (we use one standard

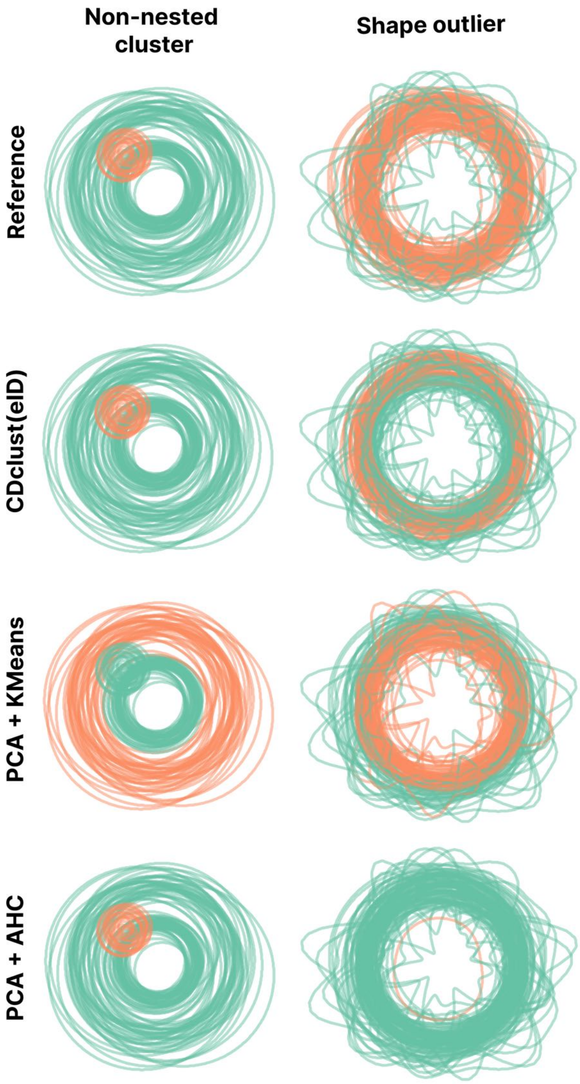


Figure 5.7.: Comparison of clustering results of CDclust, KMeans, AHC and the reference labels for the non-nested cluster and shape outlier datasets with $K = 2$.

deviation for the results in this section).

For visualization of the results, we use spaghetti plots and contour boxplots [4]. We render contour boxplots using a single hue to accommodate multiple modes of variation and consider the median (thick solid line) and the confidence bands (semi-transparent polygon with the same hue as the median line). We do not render the outliers for clarity of exposition. Nevertheless, for CDclust+ReD, we filter out the bottom ten percent of contours with the lowest depth per cluster and then compute the band with the remaining contours. We use the same computer as in the experiments with synthetic data.

5.8.1. SEGMENTATION ENSEMBLES

Data With the advent of deep learning-based auto-contouring technologies, segmentation of organs-at-risk (OARs) in radiotherapy has been largely automated [30]. Nevertheless, clinicians still need to perform a quality assessment of the segmentations, which requires understanding the uncertainty in the predictions. We consider the computerized tomography (CT) of a patient with head and neck cancer treated at HollandPTC between 2018 and 2020. The IRB approved the research protocol for the use of patient data in research, all patients signed an informed consent form. We trained 30 segmentation models based on the popular UNet architecture [31] on different subsets of the training split of the dataset of the Head and Neck Auto Segmentation MICCAI Challenge [32]. The MICCAI dataset contains CT scans of patients with head and neck cancer with ground truth segmentations of nine OARs. To further augment the ensemble size and the variability of the predictions, we trained each model using different learnable weight initializations. Using the resulting models to segment the parotid gland yields an ensemble of 120 scalar maps of per-voxel softmax probabilities. For the analysis, we focused on 540×540 pixels 2D slices of the OARs. We obtain the contours by thresholding the probabilities with an iso-value of 0.8.

Analysis Fig. 5.8 illustrates a depth-based multimodal analysis of a slice of the ensemble of segmentations of the right parotid gland using depths. We focused on the parotid gland because it is not always clearly visible in CT, which can increase inter-clinician variability. In these cases, a visual statistical summary can help clinicians understand the range of predictions. The spaghetti plot (a) provides an overview of 120 segmentations, revealing trends that are challenging to disentangle visually due to occlusion. Using contour boxplots based on the eIDs of the ensemble (b) simplifies data display and showcases variability in wide confidence bands. Notably, the median contour differs significantly from the outer band boundary, suggesting multiple modes of variation. To validate this hypothesis, we used CDclust with $K = 2$ (max average ReD). The resulting clustering reveals a split into inner and outer sections (c). The orange cluster has more members, which explains its representative shape being selected as the median in (b). While contour boxplots improve on spaghetti plots, occlusion persists. In (d), clinicians can drill down by clicking on the cluster of interest in the vertical proportions bar, revealing that most of the teal cluster's variation is concentrated in the bottom right, where confidence bands are wider.

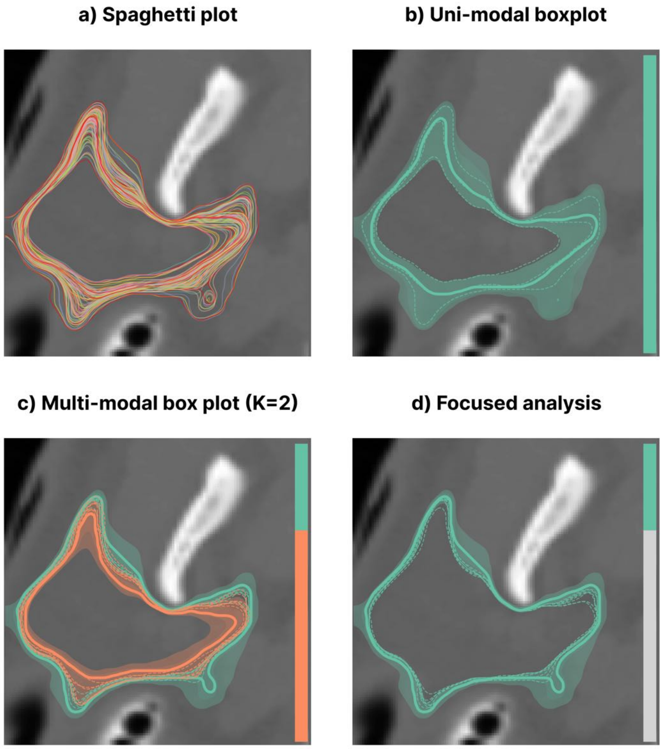


Figure 5.8.: Different stages of the ensemble analysis process. a) and b) present an overview of the ensemble using a spaghetti plot and a contour boxplot based on the depths of the complete ensemble. c) and d) present a multi-modal analysis of the ensemble. c) depicts an overview of the different modes of variation and d) focuses on the less representative variation mode.

5.8.2. WEATHER FORECAST ENSEMBLES

Data A common use case for contour statistical models is to analyze meteorological forecast data. We consider data from the European Centre for Medium-Range Weather Forecasts (ECMWF). Specifically, the ECMWF Ensemble Prediction System (EPS) provides ensembles of predictions for different variables like precipitation, temperature, and pressure. The forecasts include $N = 50$ perturbed members and a control run. We analyze the same data as in [14], which is the forecast from 00:00 UTC 15 October 2012. More details about this type of data can be found in [1]. The region under consideration encompasses 101 × 41 × 62 grid points, which corresponds to latitude, longitude, and geopotential height dimensions. For the analysis, we consider 2D fields, corresponding slices of the region where the geopotential height is $500hPa$. To obtain contours from this field, we threshold them using an iso-value of $5600m$. The spaghetti plot in Fig. 5.9 depicts the extracted contours laid over the geographical region they span.

Analysis Fig 5.9 (b) shows the results of utilizing CVP to analyze the forecast ensemble [14]. The majority of the ensemble's members belong to the purple (25) and orange (23) clusters. and the geometric medians (solid lines) are similar in shape, with the orange one exhibiting more pronounced curves towards the middle of the map. The green cluster contains the fewest members (3), and its shape differs from the other two, especially at the left of the map. When performing non-parametric analysis with CDclust (c), one can observe trends similar to CVP's. In particular, the proportions (24, 17, and 10 members) remain similar, and the shapes of the representatives too. This shows that both clustering procedures identified similar trends in the data. The two methods mainly differ in the bands' shapes and the representatives' smoothness. The depth-based bands are generally thicker, and the trajectories of the representatives are more distinct because they are made from inlier contours in the ensemble. In contrast, CVP synthesizes bands and representatives, producing smoother graphical elements. A clear visual difference that arose in this case study is the blob in CDClust's green cluster. Both methods use a threshold to define the bands' extents: unit standard deviation for CVP and keeping the top 90% contours depth-wise for CDclust. The blob arises because two members of the CDclust's green cluster (which agree with CVP) contain such a feature, but only one was flagged as an outlier and removed. This difference highlights the importance of trying different values for the threshold parameters of both methods. Finally, our results reinforce that, in practice, analysts can benefit from considering parametric and non-parametric analysis [18].

5.9. DISCUSSION AND CONCLUSIONS

Contour depth has gained prominence in non-parametric analysis across domains such as meteorological forecasting and medical image segmentation [4–6]. The efficacy of contour depth methods hinges on their scalability with increasing ensemble size. Our contributions significantly enhance existing methods by introducing a linear time algorithm for Epsilon Inclusion Depth (eID) computation. Furthermore,

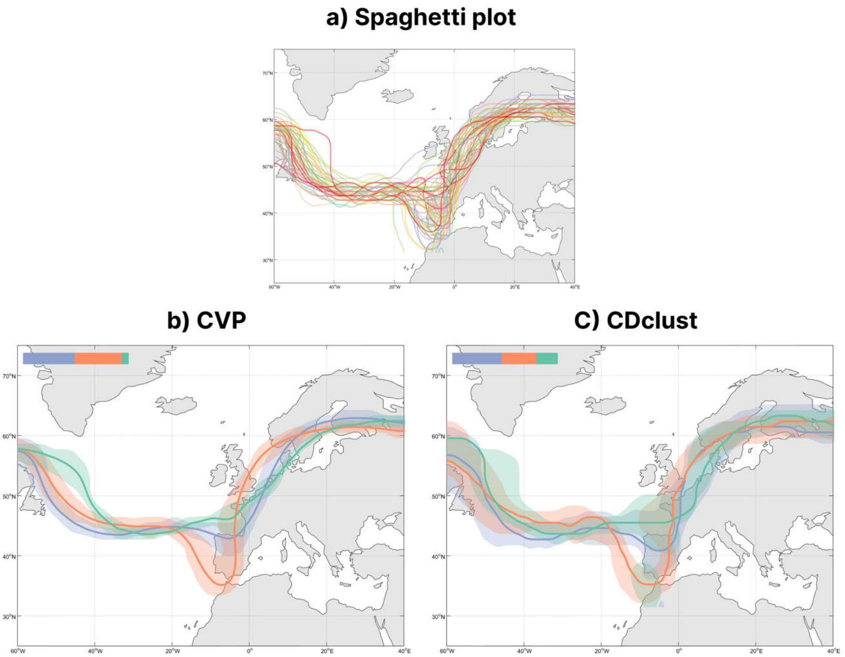


Figure 5.9.: Comparison of parametric (b) and non-parametric (c) analysis of the ensemble of 500 hPa geopotential contour lines (ECMWF ENS forecast from 00:00 UTC, 15 October 2012 valid at 00:00 UTC, 20 October 2020). (a) presents an overview of the ensemble using a spaghetti plot. The horizontal colored bar in (b) and (c) encodes the cluster's proportions in decreasing order.

we introduce an inclusion matrix, facilitating depth computation on ensemble subsets without reevaluating the inclusion relationship, a process dependent on domain resolution. These accelerated depth computation methods find applications in progressive depth computation [29] and interactive depth updating.

We also generalize contour depth using relative depth and introduce CDclust to address the assumption of contours drawn from the same distribution. To our knowledge, CDclust is the first depth-based contour clustering algorithm. Experiments on synthetic data demonstrate that CDclust largely agrees with KMeans and Agglomerative Hierarchical Clustering, but exhibits sensitivity to clusters violating the nestedness relationship. The desirability of this property depends on the application. We further demonstrated CDclust's practical utility by analyzing ensembles arising from two domains. In medical image segmentation, we showcase how clinicians can disentangle trends through multi-modal analysis. This positions contour depth methodology for interactive refinement of segmentations [33, 34] based on representative selection [5]. Our meteorological forecasting example compares non-parametric and parametric multi-modal analyses, revealing the visualization-altering assumptions of CVP's method. Adopting both parametric and non-parametric lenses is crucial in practice [18]. The proposed methods and the contour-depth Python library contribute to this approach.

There are several future work avenues. First, eID's formulation facilitates obtaining a linear algorithm based on precomputed maps. It is unclear whether other depth notions like contour band depth [4] can profit from similar strategies. Second, the runtime of linear eID depends on the grid resolution, reducing its effectiveness in cases that require multiple evaluations. Addressing this dependency and implementing parallelism, for instance, via a GPU implementation, would increase the contour depth methodology's reach. Third, using ReD to select the optimal K showed suboptimal clusters can obtain high ReD. While alternative schemes are possible, we found running CDclust multiple times helps avoiding local optima. Finally, CDclust uses a global depth notion. Future investigations could adapt CDclust to enable local analysis [5] for multi-scale insights. Additionally, working directly with the scalar field from which contours arise and integrating speedups into functional depth cases are intriguing future research avenues [22].

REFERENCES

[1] M. Leutbecher and T. Palmer. “Ensemble forecasting”. In: *Journal of Computational Physics* 227.7 (2008), pp. 3515–3539. doi: 10.1016/j.jcp.2007.02.014.

[2] E. W. Korevaar, S. J. M. Habraken, D. Scandurra, R. G. J. Kierkels, M. Unipan, M. G. C. Eenink, R. J. H. M. Steenbakkers, S. G. Peeters, J. D. Zindler, M. Hoogeman, and J. A. Langendijk. “Practical robustness evaluation in radiotherapy - A photon and proton-proof alternative to PTV-based plan evaluation”. In: *Radiotherapy and Oncology* 141 (2019), pp. 267–274. doi: 10.1016/j.radonc.2019.08.005.

[3] A. Myers and N. Miolane. “Regression-Based Elastic Metric Learning on Shape Spaces of Cell Curves”. In: *NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*. 2022. url: https://openreview.net/forum?id=8YKd0rwc4mu.

[4] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. “Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[5] M. Mirzargar and R. T. Whitaker. “Representative Consensus from Limited-Size Ensembles”. In: *Computer Graphics Forum* 37.3 (2018), pp. 13–22. doi: 10.1111/cgf.13397.

[6] N. F. Chaves-de-Plaza, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. “Inclusion Depth for Contour Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* (2024), pp. 1–12. doi: 10.1109/TVCG.2024.3350076.

[7] R. Jörnsten. “Clustering and classification based on the L1 data depth”. In: *Journal of Multivariate Analysis* 90.1 (2004), pp. 67–89. doi: 10.1016/j.jmva.2004.02.013.

[8] Y. Sun, M. G. Genton, and D. W. Nychka. “Exact fast computation of band depth for large functional datasets: How quickly can one million curves be ranked?” In: *Stat* 1.1 (2012), pp. 68–74. doi: 10.1002/sta4.8.

[9] G. Pandolfo and A. D’ambrosio. “Clustering directional data through depth functions”. In: *Computational Statistics* 38.3 (2023), pp. 1487–1506. doi: 10.1007/s00180-022-01281-w.

[10] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi. “A review of uncertainty quantification in deep learning: Techniques, applications and challenges”. In: *Information Fusion* 76 (2021), pp. 243–297. doi: 10.1016/j.inffus.2021.05.008.

[11] J. Wang, S. Hazarika, C. Li, and H.-W. Shen. “Visualization and Visual Analysis of Ensemble Data: A Survey”. In: *IEEE Transactions on Visualization and Computer Graphics* 25.9 (2019), pp. 2853–2872. doi: 10.1109/TVCG.2018.2853721.

[12] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead. “Noodles: A Tool for Visualization of Numerical Weather Model Ensemble Uncertainty”. In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 1421–1430. doi: 10.1109/TVCG.2010.181.

[13] L. Padilla, R. Fygenson, S. C. Castro, and E. Bertini. “Multiple Forecast Visualizations (MFVs): Trade-offs in Trust and Performance in Multiple COVID-19 Forecast Visualizations”. In: *IEEE Transactions on Visualization and Computer Graphics* 29.1 (2023), pp. 12–22. doi: 10.1109/TVCG.2022.3209457.

[14] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann. “Visual Analysis of Spatial Variability and Global Correlations in Ensembles of Iso-Contours”. In: *Computer Graphics Forum* 35.3 (2016), pp. 221–230. doi: 10.1111/cgf.12898.

[15] F. Ferstl, K. Bürger, and R. Westermann. “Streamline Variability Plots for Characterizing the Uncertainty in Vector Field Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 767–776. doi: 10.1109/TVCG.2015.2467204.

[16] K. Pothkow and H.-C. Hege. “Positional Uncertainty of Isocontours: Condition Analysis and Probabilistic Measures”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.10 (2011), pp. 1393–1406. doi: 10.1109/TVCG.2010.247.

[17] A. Kumpf, B. Tost, M. Baumgart, M. Riemer, R. Westermann, and M. Rautenhaus. “Visualizing Confidence in Cluster-Based Ensemble Weather Forecast Analyses”. In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), pp. 109–119. doi: 10.1109/TVCG.2017.2745178.

[18] M. Zhang, Q. Li, L. Chen, X. Yuan, and J. Yong. “EnConVis: A Unified Framework for Ensemble Contour Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* 29.4 (2023), pp. 2067–2079. doi: 10.1109/TVCG.2021.3140153.

[19] I. Demir, M. Jarema, and R. Westermann. “Visualizing the Central Tendency of Ensembles of Shapes”. In: SIGGRAPH ASIA 2016 Symposium on Visualization. Macau: ACM, 2016. isbn: 9781450345477. doi: 10.1145/3002151.3002165. url: https://doi.org/10.1145/3002151.3002165.

[20] D. Paindaveine and G. Van Bever. “From Depth to Local Depth: A Focus on Centrality”. In: *Journal of the American Statistical Association* 108.503 (2013), pp. 1105–1119. doi: 10.1080/01621459.2013.813390.

[21] D. M. Thomas and V. Natarajan. “Multiscale Symmetry Detection in Scalar Fields by Clustering Contours”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 2427–2436. doi: 10.1109/TVCG.2014.2346332.

[22] B. Ma and A. Entezari. “An Interactive Framework for Visualization of Weather Forecast Ensembles”. In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), pp. 1091–1101. doi: 10.1109/TVCG.2018.2864815.

[23] P. Lafaye de Micheaux, P. Mozharovskyi, and M. Vimond. “Depth for Curve Data and Applications”. In: *Journal of the American Statistical Association* 116.536 (2021), pp. 1881–1897. doi: 10.1080/01621459.2020.1745815.

[24] Y. Ding, X. Dang, H. Peng, and D. Wilkins. “Robust clustering in high dimensional data using statistical depths”. In: *BMC Bioinformatics* 8.7 (2007), S8. doi: 10.1186/1471-2105-8-S7-S8.

[25] M.-H. Jeong, Y. Cai, C. J. Sullivan, and S. Wang. “Data Depth Based Clustering Analysis”. In: *ACM SIGSPATIAL*. 2016. doi: 10.1145/2996913.2996984.

[26] X. Huang and Y. R. Gel. “CRAD: Clustering with Robust Autocuts and Depth”. In: *2017 IEEE International Conference on Data Mining (ICDM)*. 2017, pp. 925–930. doi: 10.1109/ICDM.2017.116.

[27] C. Patil and I. Baidari. “Estimating the Optimal Number of Clusters k in a Dataset Using Data Depth”. In: *Data Science and Engineering* 4.2 (2019), pp. 132–140. doi: 10.1007/s41019-019-0091-y.

[28] D. Sculley. “Web-Scale k-Means Clustering”. In: *Proceedings of the 19th International Conference on World Wide Web*. WWW ’10. Raleigh, North Carolina, USA: Association for Computing Machinery, 2010, pp. 1177–1178. isbn: 9781605587998. doi: 10.1145/1772690.1772862. url: https://doi.org/10.1145/1772690.1772862.

[29] C. D. Stolper, A. Perer, and D. Gotz. “Progressive Visual Analytics: User-Driven Visual Exploration of In-Progress Analytics”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 1653–1662. doi: 10.1109/TVCG.2014.2346574.

[30] E. Montagnon, M. Cerny, A. Cadrin-Chênevert, V. Hamilton, T. Derennes, A. Ilinca, F. Vandenbroucke-Menu, S. Turcotte, S. Kadoury, and A. Tang. “Deep learning workflow in radiology: a primer”. In: *Insights into Imaging* 11.1 (Feb. 10, 2020), p. 22. issn: 1869–4101. doi: 10.1186/s13244-019-0832-5.

[31] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Cham: Springer International Publishing, 2015, pp. 234–241. isbn: 978-3-319-24574-4. doi: 10.1007/978-3-319-24574-4_28.

[32] P. F. Raudaschl, P. Zaffino, G. C. Sharp, M. F. Spadea, A. Chen, B. M. Dawant, T. Albrecht, T. Gass, C. Langguth, M. Lüthi, F. Jung, O. Knapp, S. Wesarg, R. Mannion-Haworth, M. Bowes, A. Ashman, G. Guillard, A. Brett, G. Vincent, M. Orbes-Arteaga, D. Cárdenas-Peña, G. Castellanos-Dominguez, N. Aghdasi, Y. Li, A. Berens, K. Moe, B. Hannaford, R. Schubert, and K. D. Fritscher. “Evaluation of segmentation methods on head and neck CT: Auto-segmentation challenge 2015”. In: *Medical Physics* 44.5 (2017), pp. 2020–2036. doi: 10.1002/mp.12197. eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.12197. url: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12197.

[33] R. Wang, T. Lei, R. Cui, B. Zhang, H. Meng, and A. K. Nandi. “Medical image segmentation using deep learning: A survey”. In: *IET Image Processing* 16.5 (2022), pp. 1243–1267. doi: 10.1049/ipr2.12419.

[34] N. Tajbakhsh, L. Jeyaseelan, Q. Li, J. N. Chiang, Z. Wu, and X. Ding. “Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation”. In: *Medical Image Analysis* 63 (2020), p. 101693. doi: 10.1016/j.media.2020.101693.

6

LOGCC: LOCAL-TO-GLOBAL CORRELATION CLUSTERING FOR SCALAR FIELD ENSEMBLES

While Chapters 4 and 5 focus on analyzing ensemble statistics and identifying modes of variation, Chapter 6 tackles the complementary problem of investigating the ensemble's structural spatial variability (i.e., the correlation structure between spatial locations). Existing correlation clustering methods help with this task but suffer from high computational costs, making them impractical for interactive analysis of segmentation ensembles with large domains (i.e., spanning a large voxel grid), like high-resolution patient images in radiotherapy. We introduce Local-to-Global Correlation Clustering (LoGCC), a two-step correlation clustering framework that efficiently partitions ensembles into highly correlated regions. The local step identifies spatially adjacent groups of correlated voxels, while the global step reveals distant structural connections using weak transitivity properties. By uncovering spatially consistent building blocks, LoGCC provides new insights into how locations correlate across an ensemble's spatial domain. The contour depth methodology presented in the previous chapters and LoGCC form a versatile toolkit for analyzing segmentation ensembles across different dimensions.

This chapter is based on the submitted paper: Chaves-de-Plaza, Nicolas F., R. G. Raidou, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles". Submitted for publication. 2025

6.1. INTRODUCTION

Correlation clustering (CC) has emerged as a tool to analyze structural changes in scalar field ensembles that arise due to dependencies between uncertain cells at different locations [1–4]. In this context, a cell refers to the smallest unit or grid point within the scalar field ensemble's domain. CC methods shed light on local and global relationships between cells by partitioning the domain into clusters with highly correlated cells assigned to the same cluster. For instance, in the ensemble shown in Fig. 6.1 (a), some structures only appear in a subset of the members, locally correlating the cells they spawn. Other distant structures tend to appear jointly, implying global connections between cells.

Existing CC methods have significant computational costs with time and memory requirements scaling quadratically with scalar field resolution (i.e., the number of cells). This unfavorable scaling behavior has limited their applicability to two-dimensional scalar field ensembles with a cell number on the order of thousands, for which a complete CC analysis can take hours to complete [1].

Pivot-based CC methods iteratively partition the domain by selecting a random cell as the pivot and adding to its cluster all other cells with which it shares a positive relationship (i.e., correlation of the ensemble values at the two cells lies above a given threshold). This process repeats until no cells remain to be assigned [1, 5]. These methods are attractive because they are conceptually simple and avoid storing all the pairwise correlations between cells. Nevertheless, they are still significantly limited by their worst-case time complexity, which depends quadratically on scalar field resolution.

In this paper, we present the novel Local-to-Global Correlation Clustering (LoGCC) framework, which leverages the spatial structure (i.e., the neighborhood graph) of the scalar field's domain to significantly accelerate pivot-based CC methods. As depicted in Fig. 6.1 (b) and (c), LoGCC separates the CC analysis into local and global steps, respectively. In the local step, LoGCC leverages the domain's neighborhood graph to build highly correlated local clusters from randomly sampled cells in linear time. In the global step, LoGCC builds the correlation clustering for a given threshold combining the clusters formed in the local step based on their correlations.

A key contribution of the LoGCC framework is its ability to leverage weak transitivity relationships between correlations to dramatically reduce the number of cell pairs that need to be compared when merging local clusters. Assessing all possible relationships between cells in a pair of local clusters would hamper runtime gains. Instead, we focus on evaluating the relationships between the pivots of the local clusters. By exploiting the weak transitivity properties of correlations, we derive lower bounds on the possible values of correlations between clusters. Our results show that for a range of local and global correlation thresholds, LoGCC provides stricter theoretical bounds compared to the Pivot method [5].

Existing pivot-based CC algorithms can be adapted to LoGCC's two-step structure, significantly improving their performance without affecting the quality of the results. LoGCC's local step scales linearly in the scalar field resolution, and its global step exhibits the same time complexity as the adapted CC algorithm when applied to the significantly lower number of primitives the local step yields. In addition to the significant speedup per CC run, LoGCC's local step can be reused across different global thresholds, enabling more efficient exploration of multiple correlation values.

We demonstrate LoGCC's flexibility by leveraging it to speed up the practically relevant Pivot [5] and correlation neighborhood CN-Pivot [1] CC methods and evaluating the performance of the local-to-global variants on three datasets from different application domains. Furthermore, we evaluate the scaling behavior of the proposed framework using a synthetic dataset that permits modulating critical parameters like the field size and the number of clusters. The experiments show the significant performance gains LoGCC offers, which can further increase by lowering the local correlation threshold without a substantial drop in clustering quality.

In summary, the main contributions of the paper are:

- *The Local-to-Global Correlation Clustering (LoGCC) framework.* LoGCC uses the spatial structure of the domain underlying the scalar fields to first form local clusters and then merge them into global clusters. This substantially accelerates pivot-based correlation clustering and enables the efficient exploration of a wide range of correlation thresholds.
- *Weak correlation transitivity for cluster merging.* LoGCC leverages weak transitivity properties of correlation to reduce cell pair comparisons during cluster merging. This is crucial for the runtime efficiency and is based on lemmas providing lower bounds for threshold selection.
- *Adaptation of CC algorithms.* We use the LoGCC framework to accelerate the Pivot and CN-Pivot CC methods. Experimental results demonstrate substantial performance gains across datasets without sacrificing clustering quality.

6.2. RELATED WORK

Ensemble uncertainty visualization. Our work falls within the uncertainty visualization research domain, focusing specifically on the

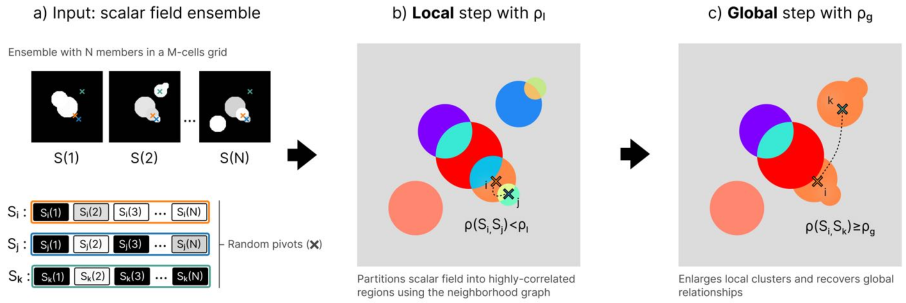

a) Input: scalar field ensemble
Ensemble with N members in a M-cells grid
S(1)
S(2)
S(N)
S_i : $S_i(1)$ $S_i(2)$ $S_i(3)$ ... $S_i(N)$
S_j : $S_j(1)$ $S_j(2)$ $S_j(3)$ ... $S_j(N)$
S_k : $S_k(1)$ $S_k(2)$ $S_k(3)$ ... $S_k(N)$
Random pivots (✖)
b) **Local** step with ρ_l
i
j
$\rho(S_i,S_j)<\rho_l$
Partitions scalar field into highly-correlated regions using the neighborhood graph
c) **Global** step with ρ_g
k
i
$\rho(S_i,S_k)\geq\rho_g$
Enlarges local clusters and recovers global relationships

Figure 6.1.: Overview of the Local-to-Global Correlation Clustering (LoGCC) framework. Taking as input an ensemble of scalar fields, whose domain has been discretized into cells (a), LoGCC first computes a local clustering based on adjacent cells' relationships (b). In the global step (c), LoGCC combines local clusters, uncovering global relationships.

visualization of ensemble data. Previous studies in this area can be categorized based on data type, dimensionality, and visualization strategies [6]. The predominant approach for analyzing scalar field ensemble uncertainty consists of examining the spread of features or variables at individual cells either directly on the scalar field [7] or, more commonly, on iso-contours derived from scalar fields [8] using parametric [9] and non-parametric models [10–14]. Our approach extends beyond cell-based analysis by considering the relationships between cells and operates directly on the scalar fields, as extracting contours can lead to structural information loss.

Correlation analysis and visualization. Several techniques address correlations between variables. Correlation maps permit analyzing pairwise relationships between large numbers of categorical and numerical variables [15]. Another strategy groups similarly distributed variables and represents these groups with a centroid [16]. For scalar field ensembles, Multifield-Graphs [17] allow visual exploration of correlation fields derived from ensemble members. This approach remains location-based, comparing values at the same cell across ensemble members rather than comparing values at different cell locations, which is essential for detecting structural changes in scalar field ensembles due to dependencies between cells [18].

Correlation clustering. Correlation clustering (CC) has emerged as a central tool for analyzing structural changes in scalar field ensembles due to dependencies between cells. CC methods are related to superpixel approaches, which partition an input image's domain into perceptually similar regions [19]. Nevertheless, they differ from superpixels in the objective choice and the type of data and representation they operate on. Existing CC methods differ in the problem formulation and their support of extensions like extracting inverse correlations and performing hierarchical clustering. One approach represents cells in a different space to enhance clustering quality. For example, correlation-based metrics can be used in nearest neighbor searches on a hyperspectral projection, producing clusters that can be explored via a dendrogram [2]. Another method maps cells to a 3D space aligned with a color space, where segmentation reveals regions of local and global cell connections [4]. While some works utilize the locality of correlations in scalar fields [18, 20], they do not leverage this to accelerate CC computation as we do.

A major challenge with existing CC methods is their high computational cost, as they require computing all pairwise correlations between scalar field cells, limiting their scalability to larger datasets (e.g., 3D ensembles). To improve efficiency, some methods use distributed and parallel processing [21], while others employ sampling techniques based on domain knowledge [22] or Bayesian optimal sampling [23]. However, these methods are confined to hierarchical grids, overlooking the anisotropic shape of correlation neighborhoods [1]. In this work, we propose a two-step domain partitioning that accounts for this anisotropy

Pivot-based correlation clustering. CC is a longstanding problem in machine learning, with recent attention in the visualization community. The typical CC problem formulation involves grouping graph vertices to minimize the number of correlated pairs separated and uncorrelated pairs clustered together—an NP-Hard problem [24]. The 3-approximation Pivot algorithm [5] is widely studied for its simplicity, ease of implementation, and practical performance. Subsequent work improved the approximation guarantee to 2 using linear integer programming [25, 26], though at the cost of quadratic runtime. Parallelization has improved speed [27, 28]. In this work, we demonstrate further speedups by leveraging spatial structure in the graph.

Pivot-based algorithms can analyze scalar field ensembles by treating cells as vertices and thresholded correlations as edges. The CN-Pivot algorithm produces clusters that reflect the anisotropy and strength of correlation neighborhoods [1], revealing both local and global relationships. Unlike other CC methods in visualization [2, 4], CN-Pivot doesn't require storing the full pairwise correlation matrix. It can also handle inverse correlation structures and hierarchical clustering. Further, it can be extended to consider correlation cliques, which form more compact clusters by accounting for all pairwise relationships within a cluster [3]. Despite these advantages, CN-Pivot, like Pivot, faces quadratic runtime challenges. We show how LoGCC can significantly accelerate CN-Pivot, enhancing its practical utility.

6.3. BACKGROUND

We consider an ensemble $S = \{S(1), S(2), \ldots, S(N)\}$ of N scalar fields on a domain in a Euclidean space. For discretization, the domain is decomposed into cells, *e.g.* by a regular grid, and each scalar field is represented by a real number per cell. We index the cells and describe each scalar field by a vector $S(n) \in \mathbb{R}^M$, which stacks the values the scalar field takes on all the cells. Here M is the number of cells. $S_i(n)$ denotes the i^{th} entry of $S(n)$ and $S_i \in \mathbb{R}^N$ the vector stacking the values $S_i(n)$ for all n. Fig. 6.1 (a) presents an example of the notation used for three cells. We describe the correlation of the ensemble at a pair of cells (i, j) by evaluating the correlation $\rho(S_i, S_j)$ of S_i and S_j. Different correlation functions ρ can be used. We use the Pearson correlation coefficient given by

$$\rho(S_i, S_j) = \frac{\sum_{n=1}^{N}(S_i(n) - \bar{S}_i)(S_j(n) - \bar{S}_j)}{\sqrt{\sum_{n=1}^{N}(S_i(n) - \bar{S}_i)^2 \sum_{n=1}^{N}(S_j(n) - \bar{S}_j)^2}}, \tag{6.1}$$

where $\bar{S}_i$ denotes the mean of the N entries of S_i.

Pivot-based CC algorithms operate on a complete undirected and unweighted graph $G = (V, E)$. Based on a given correlation threshold ρ_t, the edge set E is split into two disjoint sets: the correlated pairs of vertices, E^+, and the uncorrelated pairs of vertices, E^-. The goal is to cluster the vertices such that the sum of correlated pairs not in the same cluster and uncorrelated pairs within the same cluster is minimized [5, 24]. More formally, we represent the clustering by an array $I \in \{1, \ldots, K\}^M$, where K is the number of clusters and $I(i)$ indicates the cluster membership of vertex i. Then, the cost function

$$\begin{aligned} C(G, I) = &\sum_{(i,j)\in E^+} 1[I(i) \neq I(j)] + \\ &\sum_{(i,j)\in E^-} 1[I(i) = I(j)], \end{aligned} \tag{6.2}$$

where $1[\text{True}] = 1$ and $1[\text{False}] = 0$, is minimized.

To apply a pivot-based CC algorithm to an ensemble of scalar fields, the graph that has a vertex for each cell of the scalar fields' domain and the complete set of edges between all pairs of vertices is used. The subsets E^+ and E^- are given by $E^+ = \{(i, j) \in E : \rho(S_i, S_j) \geq \rho_t\}$ and $E^- = \{(i, j) \in E : \rho(S_i, S_j) < \rho_t\}$.

6.3.1. PIVOT ALGORITHM

Finding a clustering that minimizes Eq. 6.2 is NP-hard. A pivot-based 3-approximation algorithm was introduced in [5], which guarantees the clustering cost will be within three times the optimal solution. First, the algorithm selects a random unassigned vertex as a pivot. Then, it forms a cluster around the pivot by adding all other unassigned vertices with which it forms a correlated pair, thereby minimizing the number of intra-cluster disagreements. The algorithm repeats this pivot assignment and cluster-building steps until all vertices have been assigned. The algorithm's pseudocode can be found in the supplementary material.

The Pivot algorithm has a worst-case complexity of $\mathcal{O}(NM^2)$, assuming a linear cost for pairwise correlation evaluation. In practice, the actual runtime can be more favorable, depending on cluster sizes and the cluster building order. For instance, with only one cluster of size M, the algorithm takes linear time. However, performance deteriorates with more clusters. The worst-case scenario occurs when all clusters have a single vertex, requiring the algorithm to consider all vertex pairs, resulting in quadratic time complexity. Note that, due to its randomized nature, the algorithm does not guarantee identical clustering results in successive runs.

6.3.2. CN-PIVOT ALGORITHM

The correlation neighborhood (CN)-Pivot algorithm [1] is an alternative to the Pivot algorithm tailored to scalar field ensembles. Instead of aiming to minimize the cost function in Eq. 6.2, the CN-Pivot algorithm focuses on obtaining clusters whose sizes represent the strength of the correlation neighborhoods to which the respective pivots belong.

To achieve this, the method defines correlation neighborhoods. For a given vertex, its correlation neighborhood contains all other vertices with which it shares a positive relationship. More formally, the correlation neighborhood $\eta(i)$ of i can be defined as

$$\eta(i) = \{j : j \in V \wedge (i, j) \in E^{+}\}. \tag{6.3}$$

Instead of choosing the next pivot randomly, the algorithm selects the unassigned vertex with the largest correlation neighborhood. If the vertices in this neighborhood haven't been assigned to another cluster, they form a new cluster. If not, the vertex remains unassigned, and the algorithm moves on to the next vertex, following the same decreasing cardinality order. Like Pivot, CN-Pivot has a worst-case time complexity of $\mathcal{O}(NM^2)$. However, unlike Pivot, its performance doesn't improve in practice because determining the visiting order requires computing the correlation neighborhoods for every vertex, which also takes quadratic time. The algorithm's pseudocode is provided in the supplementary material.

6.4. LOCAL-TO-GLOBAL CORRELATION CLUSTERING

In this section, we introduce the Local-to-Global Correlation Clustering (LoGCC) framework, which leverages the inherent structure of scalar field ensembles to accelerate correlation clustering (CC). When applying pivot-based algorithms to cluster scalar field ensembles, they typically operate on a graph where each vertex represents a cell in the domain, and edges connect all pairs of vertices. This approach can be computationally expensive due to the need to evaluate every pair of cells. However, scalar field ensembles, defined over an Euclidean space, have additional structure. Clusters of interest in these ensembles tend to be spatially coherent rather than isolated points. LoGCC takes advantage of this by performing clustering in two steps: a **local** step, where only edges connecting neighboring cells are used to form spatially localized clusters, and a **global** step, where these localized clusters are merged to account for spatially non-local correlations.

6.4.1. THE LOCAL STEP

The aim of the first step of LoGCC is to form clusters connected in the domain of the scalar fields. We call these clusters local clusters and the step the local step. For this, LoGCC operates on a different graph than the pivot algorithms. While the graph also has a vertex for each cell of the domain of the scalar field, it only contains the edges that connect cells that are neighbors in the scalar field's domain. This neighbor graph is significantly reduced compared to the complete graph, as the number of edges is linear in the number of vertices while it is quadratic for the complete graph.

Algorithm 3 outlines the local step. It receives as input the spatial ensemble and a local correlation threshold ρ_l, which is used to assess pairwise cell relationships. We denote by $\mathcal{N}(i)$ the operator that returns the indices of the cells adjacent to cell i. The algorithm incrementally partitions the spatial domain by forming clusters around randomly selected pivots. Starting from a random pivot cell, neighboring correlated cells are iteratively added to form a cluster. The traversal stops when the cluster is surrounded only by uncorrelated cells or cells already assigned to other clusters. This step is repeated with a new random pivot not yet in a cluster, only considering cells not yet in a cluster. The traversal concludes when every cell is a member of a cluster.

6.4.2. THE GLOBAL STEP

The global step finalizes the clustering by processing the local partitions generated in the previous step. This involves constructing a new graph, the hypergraph, where each vertex represents a cluster from

Algorithm 3 Local LoGCC

```
Require: $S, \rho_l$                                   ▷ Ensemble, local correlation threshold
 1: $\Pi \leftarrow$ uniform random permutation of cells
 2: $\Psi = \varnothing; I = \varnothing$                                ▷ Local pivots and clusters
 3: $\mathcal{M} \leftarrow \varnothing$                                   ▷ Keep track of assigned cells
 4: while $\Pi \neq \varnothing$ do
 5:     $i \leftarrow pop(\Pi, 1)$                                      ▷ Pop first cell in $\Pi$
 6:     if $i \notin \mathcal{M}$ then
 7:         $\Psi \leftarrow \Psi \cup i; I_i \leftarrow \{i\}; \mathcal{M} \leftarrow \mathcal{M} \cup i$
 8:         $\mathcal{Q} \leftarrow \mathcal{N}(i)$                  ▷ Initialize queue with $i$'s neighbors
 9:         $\mathcal{V} \leftarrow \varnothing$                           ▷ Keep track of visited cells
10:         while $\mathcal{Q} \neq \varnothing$ do
11:             $j \leftarrow pop(\mathcal{Q}, 1)$
12:             if $j \in \mathcal{V}$ then
13:                 Skip $j$, go back to line 10
14:             end if
15:             $\mathcal{V} \leftarrow \mathcal{V} \cup j$
16:             $are\_corr \leftarrow \rho(S_i, S_j) \geq \rho_l$                  ▷ I.e., $(i,j) \in E^+$
17:             if $j \notin \mathcal{M} \wedge are\_corr$ then
18:                 $I_i \leftarrow I_i \cup j$
19:                 $\Pi \leftarrow \Pi - \{j\}; \mathcal{M} \leftarrow \mathcal{M} \cup j$
20:                 $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{k : k \in \mathcal{N}(j) \wedge k \notin \mathcal{M}\}$
21:             end if
22:         end while
23:     end if
24: end while
25: return $\Psi, I$
```

the local step, and edges connect all pairs of vertices. As with pivot algorithms (see Sec. 6.3), the edge set is divided into two subsets: E^+ for correlated cluster pairs and E^- for uncorrelated ones. A global threshold ρ_g determines the correlation between clusters. A pair of clusters is considered correlated if the correlation between their pivots exceeds ρ_g and uncorrelated otherwise, as illustrated in Fig. 6.2. In Algs. 4 and 5 the function $\rho_*(i, j)$ is used to evaluate the correlation between clusters i and j based on their pivots. Using pivots avoids costly pairwise comparisons between all cells in the clusters. It also allows us to reuse the established relationships between the pivots and other cells in their respective clusters from the local step. In the next subsection, we discuss how this approach provides a lower bound on the correlations between any pair of cells within a cluster formed during the global step.

Algorithm 4 Global LoGCC (Pivot Algorithm)

```
Require: $S, \rho_g, \Psi^l, I^l$    ▷ Ensemble, global correlation threshold, local pivots and clusters
$\Pi \leftarrow \Psi$                          ▷ Visit clusters in same order as pivots
$\Psi = \emptyset; I = \emptyset$                    ▷ Global pivots and clusters
while $\Pi \neq \emptyset$ do
    $i \leftarrow pop(\Pi, 1)$                 ▷ Pop first pivot in $\Pi$
    $\Psi \leftarrow \Psi \cup i$
    $I_i \leftarrow I_i^l$                      ▷ Initialize with cells of $i$'s local cluster
    for $j \in \Pi$ do
        $are\_corr \leftarrow \rho_*(i, j) \geq \rho_g$    ▷ $\rho_*$ compares clusters associated with pivots $i, j$
        if $are\_corr$ then
            $\Pi \leftarrow \Pi - \{j\}$
            $I_i \leftarrow I_i \cup I_j^l$          ▷ Add cells of $j$'s local cluster
        end if
    end for
end while
return $\Psi, I$
```

LoGCC can be used to accelerate different CC algorithms, resulting in respective global steps. We will discuss global steps mimicking Pivot and CN-Pivot. Alg. 4 shows our adaption of Pivot. The algorithmic structure remains unchanged, but operations for forming and comparing clusters are adapted to the hypergraph's vertices and edges. Adapting the CN-Pivot algorithm requires an additional pre-processing step for defining the visiting order based on the cardinalities of the correlation neighborhoods. The resulting global step is listed in Alg. 5.

Algorithm 5 Global LoGCC (CN-Pivot Algorithm)

```
Require: $S, \rho_g, \Psi^l, I^l$    ▷ Ensemble, global correlation threshold, local pivots and clusters
    $\eta_*(i) = I_i^l \cup \{I_j^l : j \in \Psi^l \wedge \rho_*(i,j) \geq \rho_g\}$    ▷ Correlation neighborhoods for all $i \in \Psi^l$ based on inter-cluster correlation measure $\rho_*$
    $\Pi \leftarrow$ permutation of $\Psi^l$ in decreasing order of correlation neighborhood cardinality
    $\Psi = \varnothing; I = \varnothing$    ▷ Global pivots and clusters
    while $\Pi \neq \varnothing$ do
        $i \leftarrow pop(\Pi, 1)$    ▷ Pop first vertex in $\Pi$
        $empty\_intersect \leftarrow \eta_*(i) \cap \eta_*(j) = \varnothing \forall j \in \Psi$
        if $empty\_intersect$ then
            $\Psi \leftarrow \Psi \cup i; I_i \leftarrow \eta_*(i)$    ▷ Build cluster
        end if
    end while
    return $\Psi, I$
```

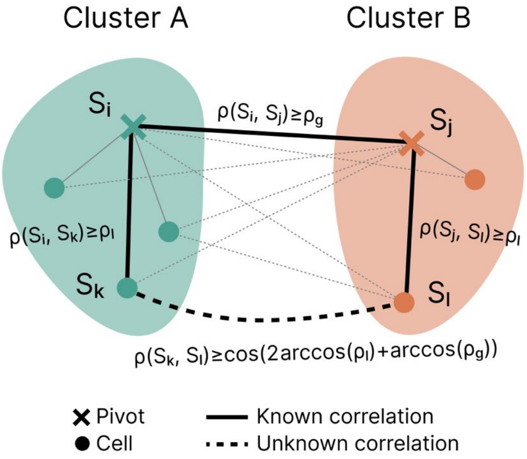

Cluster A
Cluster B
S_i
S_j
$\rho(S_i, S_j) \geq \rho_g$
$\rho(S_i, S_k) \geq \rho_l$
$\rho(S_j, S_l) \geq \rho_l$
S_k
S_l
$\rho(S_k, S_l) \geq \cos(2\arccos(\rho_l) + \arccos(\rho_g))$
Pivot
Cell
Known correlation
Unknown correlation

Figure 6.2.: Schematic of different types of relationships that arise when comparing two clusters. In LoGCC's local step, the correlations between the pivots and other cells that belong to the cluster are guaranteed to be above ρ_l. In LoGCC's global step, the correlations between the clusters' pivots are guaranteed to be above ρ_g. These two quantities determine values that between-cluster correlations can attain as stated in lemma 2.

6.4.3. PROPERTIES AND THRESHOLDS

Our design of LoGCC is based on weak transitivity properties of correlation. In particular, on two lemmas we discuss in the following. Proofs can be found in the supplementary material. Transitivity of correlation would mean that if the pairs (S_i, S_j) and (S_i, S_k) satisfy $\rho(S_i, S_j) \geq \rho_t$ and $\rho(S_i, S_k) \geq \rho_t$ for some threshold ρ_t, then also $\rho(S_j, S_k) \geq \rho_t$. This property is interesting for pivot algorithms; if S_i is a pivot of a cluster and S_j and S_k are members, then there would be a cost (see Eq. 6.2) if $\rho(S_j, S_k) < \rho_t$. Still, correlation is, in general, not transitive. Since pivot algorithms work well, however, there seems to be a possibly weaker form of correlation between S_j and S_k. The following lemma confirms this conjecture and establishes an explicit bound on the correlation of S_j and S_k.

Lemma 1 *Consider a triplet $S_i, S_j, S_k \in \mathbb{R}^N$ and assume $\rho(S_i, S_j) \geq \rho$ and $\rho(S_i, S_k) \geq \rho$ for some $\rho \in [0, 1]$. Then,*

$$\rho(S_j, S_k) \geq cos(2arccos(\rho)). \tag{6.4}$$

The lemma directly implies a property of LoGCC's local step. For a given correlation threshold ρ_t, we can choose the threshold ρ_l greater than $\cos(\arccos(\rho_t)/2)$ and guarantee that for all pairs (S_j, S_k) in each local cluster $\rho(S_j, S_k) \geq \rho_t$. This means that there are no uncorrelated pairs in any local cluster.

The second lemma is related to our global step.

Lemma 2 *Consider a quadruple $S_i, S_j, S_k, S_l \in \mathbb{R}^N$ and assume $\rho(S_i, S_j) \geq \rho_g$, $\rho(S_i, S_k) \geq \rho_l$ and $\rho(S_j, S_l) \geq \rho_l$ for some $\rho_l, \rho_g \in [0.5, 1]$. Then,*

$$\rho(S_l, S_k) \geq cos(2arccos(\rho_l) + arccos(\rho_g)). \tag{6.5}$$

In the global step, we combine clusters if the pivots of the clusters are more strongly correlated than the threshold ρ_g. As shown in Fig. 6.2, if S_k and S_l are two arbitrary elements from two local clusters with pivots S_i and S_j and the two clusters have been merged in the global step, then lemma 2 provides a lower bound for $\rho(S_l, S_k)$. This justifies our approach to evaluate the correlation of the pivots when looking for local clusters to merge.

For a target correlation threshold ρ_t, it would be possible to choose ρ_l and ρ_g such that the resulting clusters after the global step contain no uncorrelated pairs (*i.e.*, edges from E^-). In our experiments, however, this proves too strict, and we get better results with less strict values. We choose the threshold ρ_l to be larger than ρ_g. This is motivated by the fact that the value ρ_l appears in the bound from Lemma 2 with a factor of two and is confirmed by our experiments. Our experiments indicate that setting ρ_l close to one, *e.g.* $\rho_l = 0.99$, and the threshold ρ_g for the global step equal to the target threshold, $\rho_g = \rho_t$, are suitable values. Figure 6.3 shows the lower bound for the correlation of any pair of elements that are in one cluster for different target values ρ_t and the parameter setting $\rho_l = 0.99$ and $\rho_g = \rho_t$. For comparison, we also show the corresponding lower bound for Pivot, which can be derived from Lemma 1. One can see that the suggested choice of parameter values yields a stricter bound for LoGCC than Pivot for $\rho_t < 0.96$.

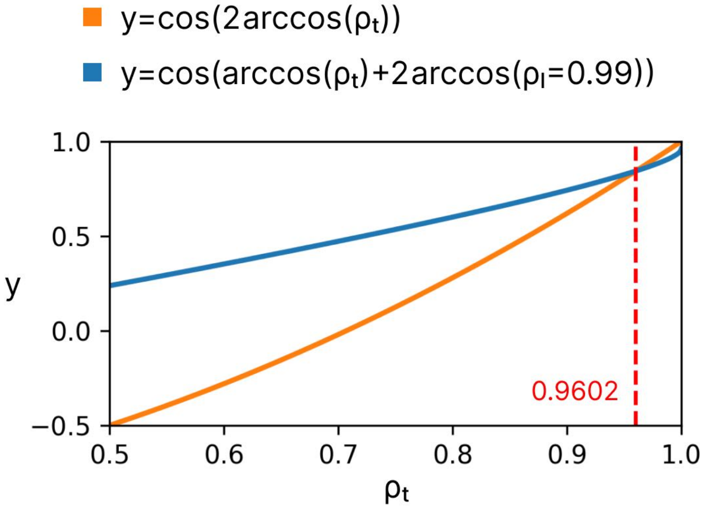


Figure 6.3.: Lower bounds for correlation of any pair of cells in one cluster for different correlation thresholds ρ_t. The orange line corresponds to the lower bound of the Pivot algorithm derived from lemma 1. The blue line corresponds to the lower bound derived in lemma 2 for correlations of pairs of cells in clusters formed in LoGCC's global step.

6.4.4. RUNTIME ANALYSIS AND EXPLORATION OF ρ_t

LoGCC's runtime is the sum of the local and global steps. In the local step in Alg. 3, each edge of the neighbor graph is visited at most twice, so its runtime scales linearly with the number of cells M. The global step's runtime is determined by the chosen algorithm applied to a hypergraph with M_h vertices and M_h^2 edges, where M_h is the number of clusters produced by the local step. Both the Pivot and CN-Pivot methods in Algs. 4 and 5 scale quadratically with the number of vertices, meaning LoGCC's global step also scales quadratically with M_h, which is a monotonically increasing function of the local correlation threshold. Our experiments show how $M_h \ll M$ for real datasets, even with high threshold values.

A single run of LoGCC has a worst-case time complexity of $\mathcal{O}(N(M+M_h^2))$, assuming a linear time cost for evaluating the correlation between N-dimensional cells. Given that $M+M_h^2 \ll M^2$, LoGCC exhibits more favorable scaling behavior than unaccelerated pivot algorithms, as we empirically validate in the experiments. Furthermore, LoGCC's decoupling of the CC process into local and global steps allows re-using the local step's output. Running LoGCC for T different correlation thresholds yields a time complexity of $\mathcal{O}(N(M+TM_h^2))$, with time savings increasing as the scalar field size grows. When analyzing multiple correlation thresholds, maintaining clustering stability is crucial for comparison. In practice, we found that using the pivot order from LoGCC's local step instead of a random order for the global step significantly enhances stability.

6.5. EXPERIMENTS AND RESULTS

In this section, we demonstrate how the LoGCC's framework can accelerate existing pivot-based CC methods without compromising clustering quality. Additionally, we demonstrate how LoGCC enables the exploration of several correlation threshold values.

6.5.1. DATA

Correlation clustering methods' outputs depend highly on the dataset to which they are applied. Therefore, we use three real datasets to demonstrate LoGCC's effectiveness: one in meteorological forecasting and two related to medical image segmentation. To further showcase the framework's scaling properties and identify data characteristics that might affect its performance, we use a synthetic dataset with adjustable parameters.

Metereological dataset (Meteo). The Meteo dataset comes from the Ensemble Prediction System (EPS) of the European Centre for Medium-Range Weather Forecasts (ECMWF). The EPS produces ensembles of predictions for several variables like pressure, temperature, and precipitation. We consider the forecast from 00:00 UTC 15 October 2012 [9]. Each forecast ensemble has fifty members obtained by varying initial conditions and an additional control run, which results in $N = 51$ members. The resulting volumetric dataset comprises $101 \times 41 \times 62$ cells. The first two dimensions correspond to latitude and longitude. The last one corresponds to slices with different geopotential height levels. We extract a 2D ensemble of scalar fields by focusing on the geopotential height level of $500hPa$, which results in members with $M = 101 \times 41 = 4141$ cells.

Medical image segmentation datasets (HaN). Ensembles of medical image segmentations can be used to model and quantify uncertainty

[29], to obtain consensus segmentations [30], and to enhance treatment plan robustness by considering multiple scenarios [31]. We use an ensemble of $N = 120$ predictions for i) the right parotid gland (ParotidR) and ii) the brain stem (BrainStem) of a head and neck patient treated at HollandPTC between 2018 and 2020. The IRB approved the research protocol for the use of patient data in research, and all patients signed an informed consent form. We refer the reader to [13] for more details about the architecture and the training process of the deep neural network (DNN) ensemble used to generate the volumetric predictions. Each member in the output ensemble is a 3D scalar field of probabilities, indicating the likelihood of a given voxel being part of the foreground. For the experiments, we focus on intermediate axial slices of the two organs, which are easier to visualize, yielding 2D scalar fields with $M = 128 \times 128$ cells. Note that, typically, DNNs' outputs are thresholded to obtain the final—binary—segmentation. We apply the CC analysis directly on the probability maps as thresholding may remove interesting structural information.

Synthetic grid (SynthGrid). To perform a fine-grained evaluation of LoGCC's scaling capabilities, we use a synthetic dataset built in the following way. We define an ensemble of N square scalar fields of dimensions $\sqrt{M} \times \sqrt{M}$, which yields M cells. Note that $Rows = Cols = \sqrt{M}$. We partition the spatial domain of the scalar field such that there are K unique local clusters. We achieve this partitioning using two mechanisms. First, we can fix the local cluster size and concatenate several clusters across the horizontal and vertical axes. This concatenation of clusters yields an increasingly larger scalar field. Second, we can start from a fixed field size and subdivide it vertically and horizontally to obtain the desired number of local clusters. In this case, increasing the number of clusters reduces their size. The first mechanism aims to simulate the scenario of analyzing larger scalar fields. For example, when switching from the Meteo to the HaN datasets. The second mechanism simulates the effect of increasing the correlation threshold value. Larger correlation thresholds lead to more and smaller local clusters. The final parameter we consider is the proportion of unique clusters, which introduces global connections. A proportion of 1 entails that every local cluster has its ID, yielding as many clusters as disconnected components in the grid. Decreasing the proportion reduces the available labels and assigns them randomly with replacement, creating global connections.

Once we partition the spatial domain using one of the approaches described before, we create the ensemble of fields in two steps. First, we select all cells with the same cluster ID and initialize them with the same N-dimensional random vector. Second, we modify the vector at each cell within the group by applying random scaling and translation transformations. Applying any CC method to this ensemble should result in a grid consisting of equally sized squares. Additional details and an example of the SynthGrid dataset are provided in the supplementary material.

6.5.2. SETUP

We compare the unaccelerated (NoAcc) and LoGCC-accelerated variants of the Pivot and CN-Pivot CC methods. For the implementation of the NoAcc variants, we followed the descriptions in Sec. 6.3 and the algorithms included in the supplementary material. The LoGCC-accelerated variants share a common local step, based on Alg. 3. We implemented custom global steps for Pivot and CN-Pivot as detailed in Algs. 4 and 5, respectively.

We ran the two variants of the Pivot and CN-Pivot CC methods for different datasets and combinations of parameters. For the real datasets, we considered $\rho_t \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. For the synthetic dataset, we varied the field size (while keeping the local cluster size fixed), the number of local clusters (while keeping the field size constant), and the proportion of unique clusters. We use smaller field sizes for the variants of the CN-Pivot method since its unaccelerated variant can take hours to complete for moderately sized ensembles. A unique combination of method-variant parameters constitutes a CC run. For each run, we save the elapsed time including local and global times for the accelerated variants and the resulting clusterings. We run ten trials per run to generate confidence intervals of the scaling and quality metrics. Unless stated otherwise, we set $\rho_g = \rho_t$ and used $\rho_l = 0.99$ for the local step of the accelerated variants. In the last experiment, we analyze the effect of lowering ρ_l on LoGCC's performance. Finally, for all experiments, we use Pearson correlation as the correlation measure. Other correlation metrics would be possible but are considered out of scope.

All the experiments in this section were run on a Mac Book Pro (2022) with an M1 Pro processor (without GPU acceleration) and 32 GB RAM. We implemented all the methods using a common Python framework [1] using Numpy vectorized operations where possible. We use optimized data structures to track which cells have been visited and added to the clusters. For the local step of the LoGCC-accelerated variants, we use an efficient queue data structure to implement the breath-first-search traversal.

6.5.3. EXPERIMENT 1: SCALING

Tab. 6.1 shows the median time of the total elapsed times in seconds for the Pivot and CN-Pivot CC methods for the three real datasets and

[1] Code can be found at https://graphics.tudelft.nl/logcc

$\rho_t \in \{0.5, 0.9\}$. The first thing to observe is how in all but one case, LoGCC-accelerated methods beat the unaccelerated counterparts. The contrast is particularly stark for the CN-Pivot algorithm, which runs in quadratic time without acceleration. The time gains are not directly evident for the accelerated Pivot method, which shows modest gains in the Meteo dataset and even loses to the unaccelerated variant when $\rho_t = 0.5$ in the HaN-ParotidR dataset. The reason for the former is that the Meteo dataset has a small field size. As for the latter, when $\rho_t = 0.5$, there are fewer and bigger clusters. Under these conditions, the runtime of the NoAcc Pivot method improves the worst-case quadratic bound. Despite Pivot scaling more favorably than CN-Pivot, results for the HaN datasets when $\rho_t = 0.9$ hint at how parameters like the field size and correlation threshold can produce a significant negative impact on the unaccelerated variants of both methods.

Table 6.1.: Median run times in seconds for the accelerated (LoGCC) and unaccelerated (NoAcc) variants of the CN-Pivot and Pivot CC algorithms.

	ρ_t	CN-Pivot[1]		Pivot[5]	
Dataset		NoAcc	LoGCC	NoAcc	LoGCC
Meteo	0.50	155.19	43.85	0.71	0.64
	0.90	154.59	43.81	5.33	3.36
HaN-ParotidR	0.50	2515.65	31.73	0.47	0.65
	0.90	2491.61	30.94	5.05	1.41
HaN-Brainstem	0.50	2496.82	39.69	0.85	0.71
	0.90	2480.31	39.10	8.52	2.03

To elucidate the scaling behavior of LoGCC, we ran several ablations for the synthetic data, including the parameters: field size, correlation threshold, and number of connected components.

Fig. 6.4 shows the results of the scaling experiment on synthetic data. The x-axis encodes the variable we increase, and the y-axis encodes the elapsed total time for each method in seconds. Note that the y-axis is logarithmic. Fig. 6.4 (a) shows the results when varying the scalar field's size. As observed, the runtime gap between variants increases as the field grows. As before, the effect is greater for CN-Pivot, which observes a speedup of more than two orders of magnitude at the maximum considered field size $\sqrt{M} = 88$. Although the gap is smaller for Pivot variants, we note that at the maximum field size $\sqrt{M} = 256$, there is more than an order of magnitude difference. In practical applications, where large fields are not rare, this performance difference makes using the unaccelerated variant hard to justify.

Fig. 6.4 (b) presents the scaling results when the number of local clusters increases, which happens when the user increases the correlation threshold ρ_t. Note that in this case, we fix the size of the field. Therefore, increasing the number of local clusters reduces their size in pixels/cells. In the case of CN-Pivot, the gap between the variants is the largest when there are fewer clusters because most of the computation will happen in LoGCC's local step, which scales linearly. As the number of local clusters increases, the number of clusters the global step receives as input also increases. In the limit, when each cell is a cluster, the performance of both variants converges. In practice, this is unlikely due to gradually fading local correlations in scalar fields. The Pivot method exhibits a slightly more complex scaling behavior due to its better scaling in practice. As it can be observed, the variants' performance tends to converge with a decreasing and increasing number of clusters. The reasoning for the latter aligns with the CN-Pivot case. As for the former, when there are few clusters, both variants will have similar runtimes because the unaccelerated version will only need to check non-relevant relationships between cells a few times.

Finally, an additional parameter that we varied in Fig. 6.4 was the proportion of unique clusters, simulating the presence of non-local connections. We observed that global connections do not affect the runtime of the CN-Pivot method. In contrast, both variants of the Pivot method see a similar systematic increase in run time. This increase occurs because Pivot methods do not revisit cells previously assigned to another cluster. Therefore, increasing the number of non-local connections has a similar effect to reducing the number of local clusters.

6.5.4. EXPERIMENT 2: CLUSTERING QUALITY

For each clustering of the real data, we compute its cost using Eq. 6.2, which quantifies the number of wrongly clustered cell pairs [24]. The line plots in Fig. 6.5 show the results for different datasets and correlation thresholds. As can be observed, there is no significant decrease in clustering quality for either CN-Pivot or Pivot. In both cases, the lines of the accelerated variants remain close to the unaccelerated ones. In some cases, we even observe that the accelerated variants attained slightly lower (better) cost function values. The CN-Pivot method attains significantly larger cost function values compared to Pivot. To our knowledge, this is the first time both methods have been experimentally compared. The results indicate that CN-Pivot's design choice of showing predominant correlation structures trades off clustering quality. Another noteworthy phenomenon is the increasing size of the confidence bands of the Pivot methods as ρ_t decreases. Upon closer inspection of the two components of the cost function, we noted this has to do with a similar increase in the variability of the cost of misassigning related cells (i.e.,

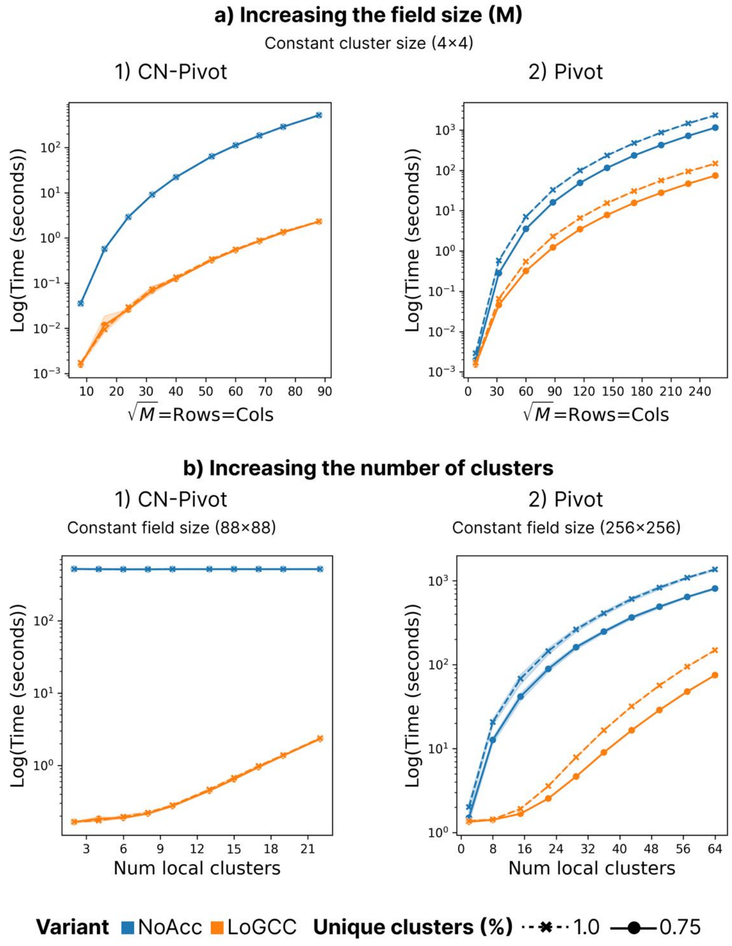


Figure 6.4.: Results of the scaling experiments on the SynthGrid dataset. We ran the CN-Pivot (1) and Pivot (2) algorithms with (orange) and without (blue) the proposed acceleration scheme for different scalar field sizes (a) and number of local clusters (b). Different line styles indicate the proportion of unique clusters in the grid (i.e., a cluster might contain multiple disconnected components). The x-axis encodes the changing variable and the y-axis encodes the time (logarithmic). The shaded areas indicate the 95% confidence interval obtained by performing ten trials for each method-variant combination per run.

positive edges across clusters).

We also qualitatively analyzed the clustering results. Fig. 6.6 presents

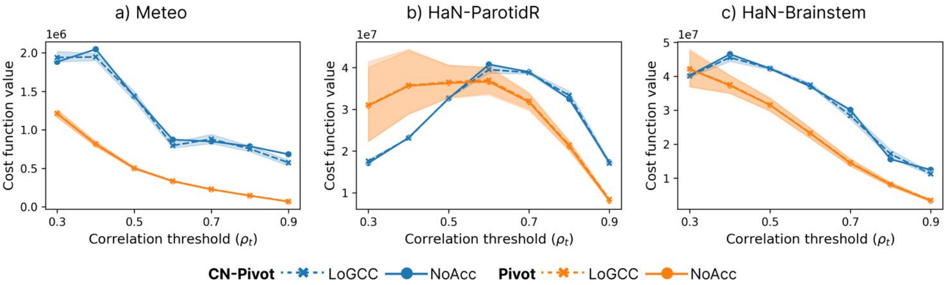


Figure 6.5.: Comparison of cost function values (Eq. 6.2) between accelerated (LoGCC) and unaccelerated (NoAcc) variants of the CN-Pivot and Pivot methods for different $\rho_t \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ for the (a) Meteo, (b) HaN-ParotidR and (c) HaN-Brainstem datasets. The shaded areas indicate the 95% confidence interval obtained by performing ten trials per run.

the outcomes for one run ($\rho_t = 0.7$) across each method-variant combination for the three real datasets. Overall, the clusterings produced by both variants are similar, with LoGCC-accelerated versions recovering clusters of comparable size and shape. This is particularly evident in the HaN datasets, where large structures, like the light and dark blue areas in HaN-Brainstem for CN-Pivot and the red area in HaN-ParotidR for Pivot, match closely. Upon closer inspection, the accelerated variants show slightly jagged cluster borders and minimal deviations in cluster composition. For instance, the red cluster in the lower-left corner of the Meteo dataset for CN-Pivot shifts slightly to the right when using the LoGCC-accelerated variant. This is due to the use of a coarser grid rather than granular scalar field cells in the global step. Lastly, the consistency between the results of the two CN-Pivot variants is noteworthy, while the Pivot results are less comparable due to the algorithm's randomized nature and sensitivity to pivot sampling.

6.5.5. EXPERIMENT 3: CORRELATION THRESHOLD EXPLORATION

By decoupling the correlation clustering algorithm into local and global steps, LoGCC enables more efficient exploration of a wide range of correlation values by reusing the results of the local step. Fig 6.7 illustrates this exploratory workflow on the HaN-ParotidR dataset using the LoGCC-accelerated CN-Pivot method. Fig. 6.7 (b) displays the sequence of global correlations derived from the same local clustering in (a). In this example, the process was eighty times faster with the accelerated variant. The unaccelerated variant would take significantly

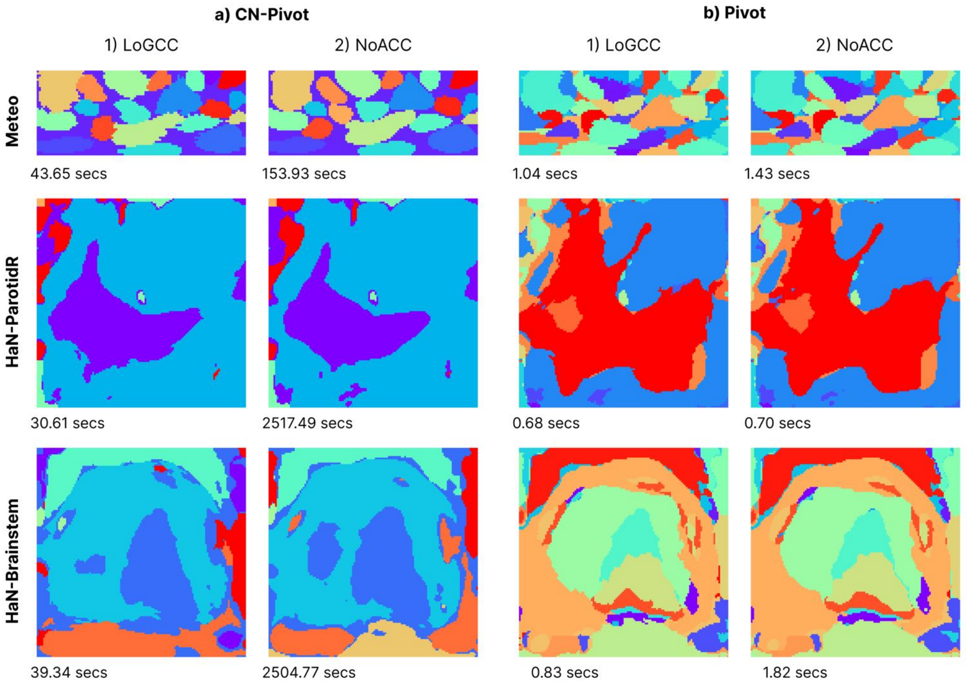


Figure 6.6.: Qualitative comparison of the clustering results of the accelerated (LoGCC) and unaccelerated (NoAcc) variants of the CN-Pivot (a) and Pivot (b) methods for the Meteo and HaN datasets using $\rho_l = 0.99$ (for the local step of accelerated variants) and $\rho_t = 0.7$.

longer, as it would require rerunning for each correlation threshold (seven runs in this case). Notably, at high correlation thresholds, the clusters closely adhere to the target segmentation boundary (dashed line). Combined with cell-wise statistics, these CC clustering results could support selection-based interactive segmentation workflows, which aim to obtain the target segmentation from information contained in the ensemble. Another example of this exploratory workflow for the HaN-Brainstem dataset is provided in the supplementary material.

Tab. 6.2 presents a more fine-grained picture of the difference in computation time between the methods and their variants. As it can be observed, the local step is relatively fast to compute because it scales linearly with the number of cells. This step only needs to be computed once and significantly reduces the number of primitives that need to be considered subsequently by the CN-Pivot and Pivot methods, which scale quadratically in the number of cells. The gap in performance for a single run gets magnified when exploring a set with several correlation thresholds, leading to significant time gains. Similarly to the single-run case, the speedups are less noticeable for the Pivot method because of its favorable scaling in practice. For CN-Pivot, we observe a similarly large gap in performance to the one observed in the single-run case. After running for the time it takes to explore seven ρ_t for the HaN-ParotidR dataset with the LoGCC-accelerated variant, a single run of the non-accelerated variant still has more than two thousand seconds to go. For both methods, this time gap grows with bigger datasets.

Table 6.2.: Median run times in seconds of the exploration process of multiple $\rho_t \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ for the accelerated (LoGCC) and unaccelerated (NoAcc) variants of the CN-Pivot and Pivot algorithms. For the accelerated variants, the table displays the times of the local and global steps separately. The total time is the sum of the median local and global times. The last column shows the speedup obtained by dividing the total times of unaccelerated and accelerated variants.

			LoGCC		NoAcc	Speedup
		Local	Global	Total	Total	
CN-Pivot	Meteo	0.23	304.70	304.93	1085.00	3.56
	HaN-ParotidR	0.60	215.26	215.86	17279.88	80.05
	HaN-Brainstem	0.61	269.32	269.93	17368.03	64.34
Pivot	Meteo	0.23	7.03	7.25	11.79	1.63
	HaN-ParotidR	0.59	1.43	2.02	9.34	4.62
	HaN-Brainstem	0.61	2.53	3.14	16.30	5.19

6.5.6. Experiment 4: LoGCC's sensitivity to decreasing ρ_l

Similarly to the unaccelerated CN-Pivot and Pivot, a bottleneck of the accelerated variants is the number of clusters the global step receives as input. So far, we used a conservative high threshold of $\rho_l = 0.99$ for the local step, which ensures the transitivity between correlation holds. High thresholds typically lead to smaller clusters, though, which increase the accelerated methods' runtimes. In this section, we explore how sensitive LoGCC is to lower values of this threshold.

Fig. 6.8 shows the results of the sensitivity study for the correlation clustering of the HaN-BrainStem dataset using the LoGCC-accelerated CN-Pivot algorithm. In (a), it can be observed how lowering ρ_l from 0.99 to 0.93 results in a dramatic time gain of another order of magnitude. The time gains between subsequent values of ρ_l are less significant, which suggests that further reducing ρ_l does not significantly reduce the number of local clusters. Regarding the quality of the clusterings, (b) shows the cost function values for different ρ_l. In general, the quality seems to be comparable. Interestingly, high ρ_l runs perform slightly better with lower ρ_t, while the trend reverses for higher ρ_t. Additionally, we observe unexpected outcomes like the winner combination $\rho_l = 0.7, \rho_t = 0.7$, which may indicate the presence of a local optima in the cost function. In (c), we analyze the results qualitatively by focusing on $\rho_t = 0.7$. As can be observed, using $\rho_l = 0.99$ leads to clusterings with more clusters and more detailed contours. Lowering ρ_l leads to fewer and coarser clusters in the local step, which greatly increases computation speed but leads to a loss of detail. For instance, only the largest correlation structures can be differentiated when $\rho_l = 0.7$. Smaller structures have been "absorbed" by larger clusters. We observed similar trends across datasets and methods. In summary, results suggest that LoGCC clusterings are robust to reductions in ρ_l. Although reducing ρ_l can lead to significant speedups, it is necessary to consider the loss of quality that inevitably comes when transitivity relationships between correlations become less informative.

6.6. DISCUSSION AND CONCLUSION

In this work, we introduced the Local-to-Global Correlation Clustering (LoGCC) framework, which accelerates pivot-based CC techniques by leveraging the inherent spatial structure of scalar fields. LoGCC's local step scales linearly with the number of cells in the scalar field, resulting in significant reductions in computational overhead. The global step, while maintaining the same scaling properties as the adapted CC algorithm, operates on a reduced set of primitives generated by the local step, thus enabling substantial speed-ups without compromising the quality of the resulting clusters. Our experiments demonstrate that LoGCC can significantly accelerate existing pivot-based CC methods like Pivot or CN-Pivot, preserving quality. Based on the derived theoretical lower bounds for correlations between clusters and empirical results, we recommend using high local correlation thresholds ($\rho_l \geq 0.96$), which yield speed-ups and minimize errors.

LoGCC's ability to reuse local step results across multiple runs of the global step offers significant computational savings, especially when exploring various correlation thresholds. In the experiments, we observed speed-ups of up to 80×, which increase with larger scalar fields. This efficiency is particularly beneficial in scenarios like user-driven meteorological analysis [3] or connectomics[32], where precomputing the local step and determining global relationships on demand can enhance workflow efficiency. Additionally, LoGCC's two-step design enables hierarchical clustering [1], ensuring consistency in cluster memberships across different global correlation thresholds, further enhancing its utility in complex applications.

Looking ahead, there are several avenues to enhance LoGCC. While we used the Pearson correlation coefficient in this study, LoGCC can adapt to other correlation measures (e.g., Kendall’s Tau [33]) if they support the transitivity required for error bounding. Our worst-case error analysis could also be refined with tighter bounds for a more accurate theoretical performance estimate. Future research could explore applying other pivot-based heuristics, such as correlation cliques [3], extending LoGCC to higher-dimensional or spatio-temporal scalar field ensembles, or different data types like polygonal meshes. Additionally, accelerating LoGCC’s local [34] and global [27] steps for real-time ensemble analysis remains a promising direction.

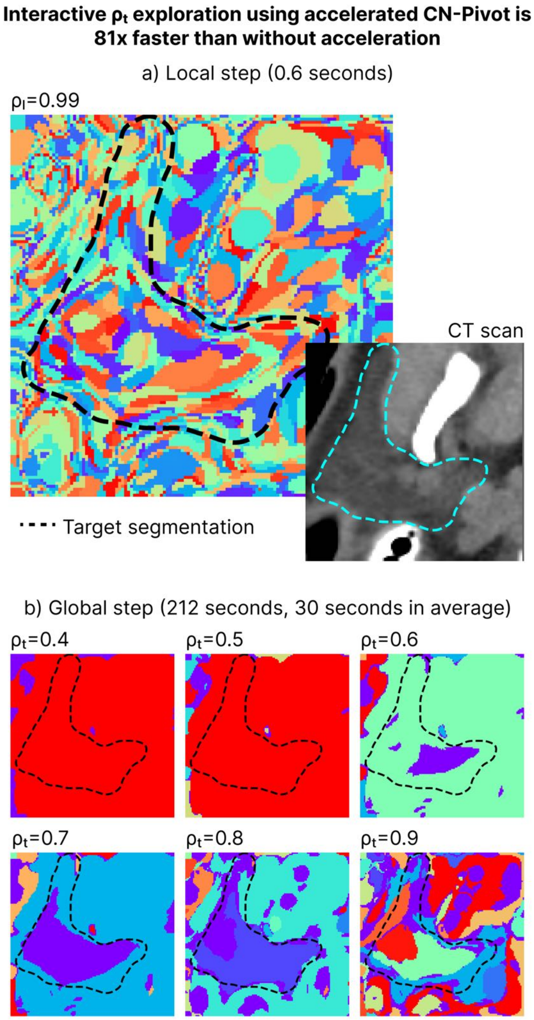

Interactive ρt exploration using accelerated CN-Pivot is 81x faster than without acceleration
a) Local step (0.6 seconds)
ρl=0.99
CT scan
Target segmentation
b) Global step (212 seconds, 30 seconds in average)
ρt=0.4
ρt=0.5
ρt=0.6
ρt=0.7
ρt=0.8
ρt=0.9

Figure 6.7.: Using the accelerated CN-Pivot method to explore the correlation structure of the HaN-ParotidR dataset using several correlation thresholds ($\rho_t \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$). The results of the local step (a) are re-used to compute several global clusterings (depicted $\rho_t \in [0.4, 0.9]$) in a fraction of the time (b). It takes the accelerated CN-Pivot method only 212 seconds to generate all the clusterings. In contrast, it takes the unaccelerated variant 17274 seconds, representing an 81× speedup. For context, we provide the CT scan slice for which the segmentation ensemble was computed and the target, ground truth, segmentation as a dashed line.

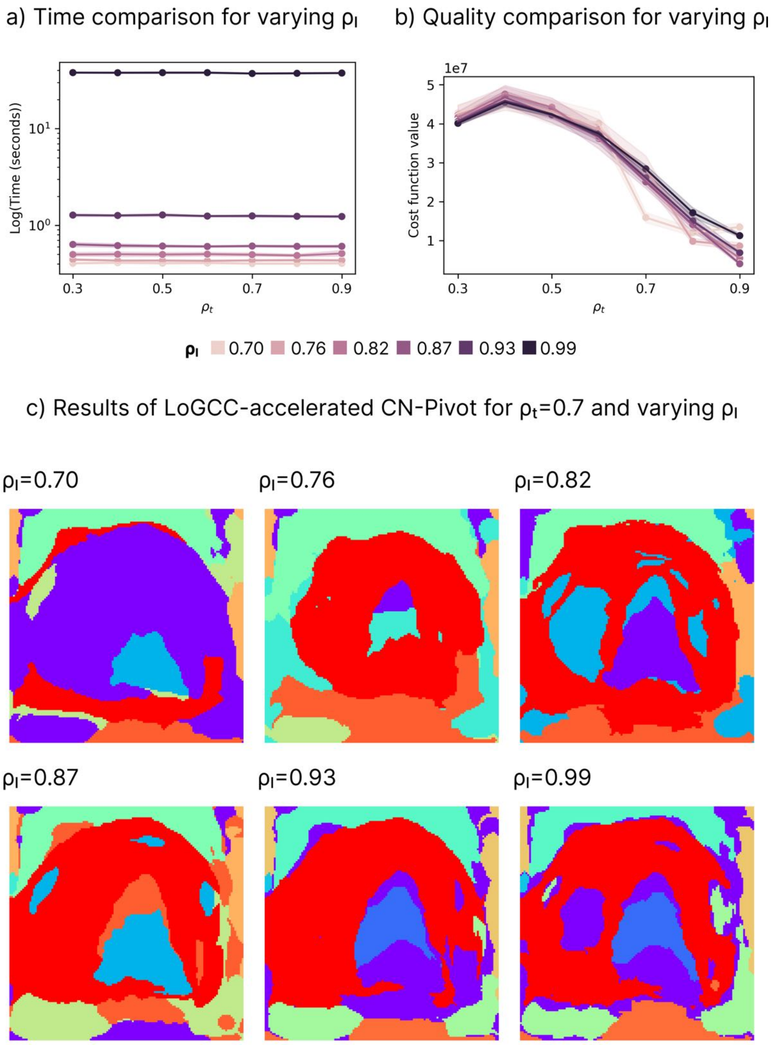


Figure 6.8.: Results of ablation study on the sensitivity of LoGCC to variations of ρ_l for the HaN-Brainstem dataset using the CN-Pivot method. (a) shows the times it took to run each ρ_l/ρ_t combination, and (b) presents the clustering's cost according to Eq. 6.2. We connect points if they share ρ_l, which varies uniformly between 0.7 and 0.99. The shaded areas indicate the 95% confidence interval obtained by performing ten trials per run. (c) presents a qualitative comparison of clustering results for $\rho_t = 0.7$ for the different ρ_l.

REFERENCES

[1] T. Pfaffelmoser and R. Westermann. “Visualization of Global Correlation Structures in Uncertain 2D Scalar Fields”. In: *Comput. Graph. Forum* 31.3pt2 (2012), pp. 1025–1034.

[2] T. Liebmann, G. H. Weber, and G. Scheuermann. “Hierarchical Correlation Clustering in Multiple 2D Scalar Fields”. In: *Comput. Graph. Forum* 37.3 (2018), pp. 1–12.

[3] A. Kumpf, M. Rautenhaus, M. Riemer, and R. Westermann. “Visual Analysis of the Temporal Evolution of Ensemble Forecast Sensitivities”. In: *IEEE Trans. Vis. Comput. Graph.* 25.1 (2019), pp. 98–108. doi: 10.1109/TVCG.2018.2864901.

[4] M. Evers, K. Huesmann, and L. Linsen. “Uncertainty-aware Visualization of Regional Time Series Correlation in Spatio-temporal Ensembles”. In: *Comput. Graph. Forum* 40.3 (2021), pp. 519–530. doi: 10.1111/cgf.14326.

[5] N. Ailon, M. Charikar, and A. Newman. “Aggregating inconsistent information: Ranking and clustering”. In: *J. ACM* 55.5 (2008). issn: 0004-5411. doi: 10.1145/1411509.1411513.

[6] J. Wang, S. Hazarika, C. Li, and H.-W. Shen. “Visualization and Visual Analysis of Ensemble Data: A Survey”. In: *IEEE Trans. Vis. Comput. Graph.* 25.9 (2019), pp. 2853–2872. doi: 10.1109/TVCG.2018.2853721.

[7] K. Pöthkow and H.-C. Hege. “Nonparametric Models for Uncertainty Visualization”. In: *Comput. Graph. Forum* 32.3pt2 (2013), pp. 131–140.

[8] M. Zhang, Q. Li, L. Chen, X. Yuan, and J.-H. Yong. “EnConVis: A Unified Framework for Ensemble Contour Visualization”. In: *IEEE Trans. Vis. Comput. Graph.* (2022), pp. 1–1. doi: 10.1109/TVCG.2021.3140153.

[9] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann. “Visual Analysis of Spatial Variability and Global Correlations in Ensembles of Iso-Contours”. In: *Comput. Graph. Forum* 35.3 (2016), pp. 221–230. doi: https://doi.org/10.1111/cgf.12898.

[10] T. Athawale, E. Sakhaee, and A. Entezari. “Isosurface Visualization of Data with Nonparametric Models for Uncertainty”. In: *IEEE Trans. Vis. Comput. Graph.* 22.1 (2016), pp. 777–786. doi: 10.1109/TVCG.2015.2467958.

[11] K. Pothkow and H.-C. Hege. “Positional Uncertainty of Isocontours: Condition Analysis and Probabilistic Measures”. In: *IEEE Trans. Visual Comput. Graphics* 17.10 (2011), pp. 1393–1406. doi: 10.1109/TVCG.2010.247.

[12] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. “Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles”. In: *IEEE Trans. Vis. Comput. Graph.* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[13] N. F. Chaves-de-Plaza, P. P. Mody, M. Staring, R. v. Egmond, A. Vilanova, and K. Hildebrandt. “Inclusion Depth for Contour Ensembles”. In: *IEEE Trans. Vis. Comput. Graph.* 30.9 (2024), pp. 6560–6571. doi: https://doi.org/10.1109/TVCG.2024.3350076.

[14] N. Chaves-de-Plaza, M. Molenaar, P. Mody, M. Staring, R. van Egmond, E. Eisemann, A. Vilanova, and K. Hildebrandt. “Depth for Multi-Modal Contour Ensembles”. In: *Comput. Graph. Forum* 43.3 (2024), e15083.

[15] Z. Zhang, K. T. McDonnell, E. Zadok, and K. Mueller. “Visual Correlation Analysis of Numerical and Categorical Data on the Correlation Map”. In: *IEEE Trans. Vis. Comput. Graph.* 21.2 (2015), pp. 289–303. doi: 10.1109/TVCG.2014.2350494.

[16] A. Biswas, S. Dutta, H.-W. Shen, and J. Woodring. “An Information-Aware Framework for Exploring Multivariate Data Sets”. In: *IEEE Trans. Vis. Comput. Graph.* 19.12 (2013), pp. 2683–2692. doi: 10.1109/TVCG.2013.133.

[17] N. Sauber, H. Theisel, and H.-p. Seidel. “Multifield-Graphs: An Approach to Visualizing Correlations in Multifield Scalar Data”. In: *IEEE Trans. Vis. Comput. Graph.* 12.5 (2006), pp. 917–924. doi: 10.1109/TVCG.2006.165.

[18] T. Pfaffelmoser and R. Westermann. “Correlation visualization for structural uncertainty analysis”. In: *Int. J. Uncertain. Quantif.* 3.2 (2013).

[19] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. “SLIC Superpixels Compared to State-of-the-Art Superpixel Methods”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.11 (2012), pp. 2274–2282. doi: 10.1109/TPAMI.2012.120.

[20] M. Berenjkoub, R. O. Monico, R. S. Laramee, and G. Chen. “Visual Analysis of Spatia-temporal Relations of Pairwise Attributes in Unsteady Flow”. In: *IEEE Trans. Vis. Comput. Graph.* 25.1 (2019), pp. 1246–1256. doi: 10.1109/TVCG.2018.2864817.

[21] Y. Su, G. Agrawal, J. Woodring, A. Biswas, and H.-W. Shen. “Supporting correlation analysis on scientific datasets in parallel and distributed settings”. In: *Proc. 23rd Int. Symp. High-Perform. Parallel Distrib. Comput.* HPDC ’14. Vancouver, BC, Canada: Association for Computing Machinery, 2014, pp. 191–202. isbn: 9781450327497. doi: 10.1145/2600212.2600230.

[22] C.-K. Chen, C. Wang, K.-L. Ma, and A. T. Wittenberg. “Static correlation visualization for large time-varying volume data”. In: *IEEE PacificVis*. 2011, pp. 27–34. doi: 10.1109/PACIFICVIS.2011.5742369.

[23] C. Neuhauser, J. Stumpfegger, and R. Westermann. “Adaptive Sampling of 3D Spatial Correlations for Focus+Context Visualization”. In: *IEEE Trans. Vis. Comput. Graph.* 30.02 (Feb. 2024), pp. 1608–1623. issn: 1941-0506. doi: 10.1109/TVCG.2023.3326855.

[24] N. Bansal, A. Blum, and S. Chawla. “Correlation Clustering”. In: *Mach. Learn.* 56.1 (2004), pp. 89–113.

[25] S. Chawla, K. Makarychev, T. Schramm, and G. Yaroslavtsev. “Near Optimal LP Rounding Algorithm for CorrelationClustering on Complete and Complete k-partite Graphs”. In: *ACM Symp. Theory Comput.* Portland, Oregon, USA: ACM, 2015, pp. 219–228. isbn: 9781450335362. doi: 10.1145/2746539.2746604.

[26] V. Cohen-Addad, E. Lee, and A. Newman. “Correlation Clustering with Sherali-Adams”. In: *Proc. - Annu. IEEE Symp. Found. Comput. Sci. FOCS*. IEEE, 2022, pp. 651–661. doi: 10.1109/FOCS54457.2022.00068.

[27] X. Pan, D. Papailiopoulos, S. Oymak, B. Recht, K. Ramchandran, and M. I. Jordan. “Parallel Correlation Clustering on Big Graphs”. In: *NeurIPS*. Vol. 28. 2015.

[28] S. Behnezhad, M. Charikar, W. Ma, and L. Tan. “Almost 3-Approximate Correlation Clustering in Constant Rounds”. In: *Proc. - Annu. IEEE Symp. Found. Comput. Sci. FOCS*. IEEE, 2022, pp. 720–731. doi: 10.1109/FOCS54457.2022.00074.

[29] P. P. Mody, N. Chaves-de-Plaza, K. Hildebrandt, R. van Egmond, H. de Ridder, and M. Staring. “Comparing Bayesian models for organ contouring in head and neck radiotherapy”. In: *Proc. SPIE Med. Imaging*. Vol. 12032. SPIE, 2022, 120320F. doi: 10.1117/12.2611083.

[30] M. Mirzargar and R. T. Whitaker. “Representative Consensus from Limited-Size Ensembles”. In: *Comput. Graph. Forum* 37.3 (2018), pp. 13–22.

[31] E. W. Korevaar, S. J. Habraken, D. Scandurra, R. G. Kierkels, M. Unipan, M. G. Eenink, R. J. Steenbakkers, S. G. Peeters, J. D. Zindler, M. Hoogeman, and J. A. Langendijk. “Practical robustness evaluation in radiotherapy – A photon and proton-proof alternative to PTV-based plan evaluation”. In: *Radiother. Oncol.* 141 (2019), pp. 267–274.

[32] A. K. Ai-Awami, J. Beyer, D. Haehn, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger. “NeuroBlocks – Visual Tracking of Segmentation and Proofreading for Large Connectomics Projects”. In: *IEEE Trans. Vis. Comput. Graph.* 22.01 (2016), pp. 738–746. issn: 1941-0506. doi: 10.1109/TVCG.2015.2467441.

[33] G. J. Székely, M. L. Rizzo, and N. K. Bakirov. “Measuring and testing dependence by correlation of distances”. In: *Ann. Stat.* 35.6 (2007), pp. 2769–2794.

[34] H. Liu, H. H. Huang, and Y. Hu. “iBFS: Concurrent Breadth-First Search on GPUs”. In: *ACM SIGMOD Int. Conf. Manag. Data*. SIGMOD ’16. ACM, 2016, pp. 403–416. isbn: 9781450335317. doi: 10.1145/2882903.2882959.

7

APPLYING CONTOUR DEPTH ANALYSIS TO REAL-WORLD RADIOTHERAPY CHALLENGES

Chapter 7 shows the potential practical utility of the contour depth methodology developed in Chapters 4 and 5 by applying it to real-world segmentation ensemble analysis workflows in radiotherapy. We consider the Contouring Collaborative for Consensus in Radiation Oncology challenge, which assessed whether non-expert segmentation ensembles could serve as an alternative to expert-derived segmentations for training artificial intelligence models. We show how depth can support and improve the analysis, comparison, and understanding of the challenge's expert and non-expert segmentation ensembles. By leveraging Inclusion Depth and Multi-Modal Contour Depth, we conduct robust and uncertainty-aware analyses, extracting insights into segmentation reliability and identifying critical variation modes within ensembles. The latter could be harnessed to power candidate selection workflows for ensemble-supported interactive segmentation refinement in quality assessment. This chapter emphasizes how contour depth techniques can enhance clinician-driven workflows and decision-making, highlighting their potential for broader adoption in clinical and research contexts.

7.1. INTRODUCTION

There is growing interest in leveraging segmentation ensembles for radiotherapy applications. Segmentation ensembles offer unique advantages, such as quantifying uncertainty and considering multiple scenarios during dose optimization. However, their adoption in clinical practice remains limited due to the challenges posed by their complexity, including the large number of members and intricate shapes they might exhibit.

This chapter shows how techniques introduced in this dissertation can help tame segmentation ensembles' complexity, increasing their usability in clinical practice. In particular, we illustrate how contour depth methodology permits robust, uncertainty-aware, and interactive analysis of segmentation ensembles in radiotherapy [1, 2]. We focus on two distinct advantages of contour depth.

First, we show how depth-based analysis can improve the robustness of downstream tasks that use the ensemble like consensus extraction. On the one hand, contour depths quantify each member's centrality within the ensemble, yielding scores that can be used to identify and exclude outliers. On the other hand, depth scores can be used to characterize the ensemble's variability through depth-sorted confidence bands, which are critical in safety-critical scenarios like radiotherapy.

Second, we explore the potential of multi-modal contour depth, which permits extracting representative candidates from the ensemble. Representative shape extraction provides an additional granularity level, which deepens the understanding of the ensemble's trends in analysis tasks. Intriguingly, this functionality also has the potential to increase the clinical usability of upcoming deep ensemble AIs. Trained on diverse data, these AIs tend to produce equally diverse predictions. Empowering clinicians to visualize representative shapes and select the desirable ones can prevent expensive edits, sparing time and effort.

We illustrate the use of contour depth methods and their advantages for radiotherapy using data from the Contouring Collaborative for Consensus in Radiation Oncology (C3RO) challenge [3]. The challenge's objective was to evaluate whether ensembles of non-expert segmentations, which are cost-effective and widely available, could replace expert ensembles for training deep ensemble AIs. This was achieved by comparing the consensuses of expert and non-expert ensembles across various metrics of geometric similarity. While results varied by region of interest, the challenge demonstrated that non-experts could approximate experts when a sufficiently large sample size was used.

The consensus-centered approach outlined above is sensitive to outliers and disregards important information of the ensemble like its variability. We revisit the research question of the C3RO challenge, re-analyzing the data with the support of the contour depth methodology. Additionally, we further the study's results by analyzing the ensembles' multi-modal structures and assessing the viability of extracted modes of variation for supporting candidate selection workflows.

We hope this study guides practitioners' usage of the contour depth methodology and inspires further research for its implementation in time-constrained contexts like radiotherapy where robustness and uncertainty-awareness are crucial.

7.2. METHODS AND MATERIALS

7.2.1. DATA

We utilize data from the Contouring Collaborative for Consensus in Radiation Oncology (C3RO) challenge [3]. The primary objective of this challenge was to evaluate whether segmentation ensembles created by non-experts could effectively replace those generated by experts for training auto-segmentation AI models. To address this question, the challenge gathered segmentation data for various regions of interest (ROIs) across five anatomical sites, assessing the agreement between expert and non-expert ensembles.

For computational efficiency, our analyses focus on the portion of the domain contained within the bounding box encompassing the union of expert and non-expert segmentation ensembles.

Table 7.1 provides a detailed summary of the C3RO dataset. The dataset includes a diverse mix of target volumes and organs-at-risk, exhibiting significant variability in size and complexity. Among the anatomical sites, the GI (gastrointestinal) and GYN (gynecological) regions contain the largest structures, as reflected in their greater slice span and volume. The H&N (head and neck) region contains more ROIs than the other regions. Finally, the number of available segmentations varies considerably across ROIs and expertise levels. For instance, non-expert segmentations are drastically more prevalent than expert segmentations, and the Breast dataset includes non-expert ensembles that are larger compared to the other anatomical regions.

7.2.2. CONSENSUS EXTRACTION AND SIMILARITY METRICS

To compare the agreement of a pair of ensembles (e.g., expert versus non-expert ensembles), we first extract a consensus segmentation from each ensemble and then compare the consensuses using the Dice Similarity Coefficient, a metric of geometric similarity that considers the amount of overlap between two segmentation masks [4, 5].

For the extraction of the consensus, we consider two schemes. First is majority voting, a simple algorithm that assigns a voxel the value that occurred most frequently in the ensemble members. We also consider the Simultaneous Truth and Performance Level Estimation (STAPLE) algorithm [6]. The STAPLE algorithm is a probabilistic consensus

Table 7.1.: Summary of regions of interest (ROIs) in the C3RO dataset. The table categorizes ROIs as either organs-at-risk (OARs) or target volumes (TV) across five anatomical regions: Breast, GI (gastrointestinal), GYN (gynecological), H&N (head and neck), and Sarcoma. For each ROI, the number of segmentation ensemble members (expert and non-expert), the mean segmentation volume (in cm^3) of experts and non-experts, and the number of axial slices in the extracted bounding box are reported.

			Number of Ensemble Members		Mean Volume (cm^3)		
Anatomical Region	ROI Type	ROI	Expert	Non-Expert	Expert	Non-Expert	Bounding Box Axial Slices
Breast	OAR	A_LAD	7	88	5	12	65
		BrachialPlex_L	6	88	12	11	50
		Heart	7	121	586	595	66
	TV	CTV_Ax	8	115	116	95	100
		CTV_Chestwall	8	117	510	463	102
		CTV_IMN	8	118	11	7	71
		CTV_Sclav_LN	8	119	36	33	39
GI	OAR	Bag_Bowel	4	23	895	1225	139
	TV	CTV_4500	4	25	1220	938	112
		CTV_5400	4	23	151	149	100
GYN	OAR	Bowel_Small	4	35	2886	1902	104
	TV	CTVn_4500	5	40	526	389	129
		CTVp_4500	5	41	281	222	88
		GTVn	5	42	8	12	89
H&N	OAR	Brainstem	13	58	26	30	30
		Glnd_Submand_L	13	57	10	10	25
		Glnd_Submand_R	12	52	12	11	27
		Larynx	12	57	36	38	28
		Musc_Constrict	11	43	26	20	51
		Parotid_L	13	59	38	32	23
		Parotid_R	13	58	40	36	23
	TV	CTV1	9	45	139	152	58
		CTV2	9	49	319	245	62
		GTVn	13	60	33	35	82
		GTVp	14	59	29	33	32
Sarcoma	OAR	Genitals	4	51	79	67	36
	TV	CTV	5	49	253	212	228
		GTV	5	60	29	26	145

extraction method that yields a probabilistic mask that optimizes the sensitivity and specificity parameters of each ensemble member. Compared to majority voting, STAPLE can perform better as it can remove random noise by reducing outlier members' contributions via weighting. A threshold is applied to obtain the final binary mask. We used the same 0.95 threshold value as in the C3RO study [3].

7.2.3. DEPTH COMPUTATION AND VISUALIZATION

We use the previously introduced Inclusion Depth (ID) and multi-modal ID notions for uni-modal and multi-modal contour depth analysis, respectively [1, 2]. In all cases, we use the relaxed Epsilon Inclusion Depth (eID) as it better handles real-world datasets where contours are not nested (i.e., their boundaries have multiple intersections).

Based on the depth scores, we differentiate between inlier and outlier segmentation sets. In the uni-modal case, we discard a percentage of the segmentations with the lowest depths, the outliers. We tried several percentages and found that trimming thirty percent of the outliers leads to robust downstream estimators without shrinking the inlier set too much. Note one can use other heuristics or automatic methods to determine this trimming percentage. For the multi-modal analyses, we chose not to discard outliers because trimmed variation modes have too few members for reliable consensus extraction and uncertainty visualization tasks.

We extract the robust consensuses by applying the algorithms described in the previous subsection to the inlier sets. For visual analysis, we use simplified contour boxplots that use a solid line to indicate the ensemble's consensus and a shaded area to indicate the region that the top fifty percent of the segmentations, depth-wise, occupy. We refer to this area as the fifty percent confidence band.

7.3. RESULTS

In this section, we illustrate the use of the contour depth techniques developed in this thesis for re-analyzing the C3RO challenge dataset. We start by demonstrating how contour depths can enhance the robustness of segmentation ensemble analysis methods and then explore their multi-modal capabilities, which could potentially enable candidate selection workflows.

7.3.1. ROBUST AND UNCERTAINTY-AWARE CONSENSUS EXTRACTION USING INCLUSION DEPTH

To recapitulate, the main question of the C3RO challenge was to assess whether non-experts can approximate the behavior of expert ensembles. We address re-visit this question, leveraging the depth methodology for robust analysis. Specifically, we compare the geometric similarity of the STAPLE consensus of the expert ensemble (the desired ground truth segmentation) against the consensuses of the non-expert ensemble obtained with different methods and subsets of the data. Figure 7.1 shows the DSC results for the different ROIs and anatomical regions of the C3RO dataset. Teal and orange lines indicate expert STAPLE comparison against non-expert consensus obtained using majority voting and STAPLE. Dashed and solid lines indicate that full and depth-trimmed ensembles were used for the consensus extraction.

The first thing to observe is how STAPLE yields higher agreeing consensuses than majority voting. Only in some cases like the CTV_5400 (GI) and the Brainstem (H&N) do the majority voting methods yield a similar or better agreement. The edge of STAPLE over majority voting occurs for two reasons. First, we use the STAPLE of the expert ensemble, the gold standard in the C3RO study, as the reference consensus for all comparisons. Second, STAPLE and majority voting performances differ, with the latter being more sensitive to outlying segmentations.

A central result is how, regardless of the consensus extraction algorithm, using depth to trim outliers improves the agreement between experts and non-experts. This demonstrates that shape outlier removal can lead to robustness gains and new insights. In the case of the C3RO dataset, comparisons using the depth-trimmed ensembles evidence increased agreement between experts and non-experts, further endorsing the usage of the latter. Nevertheless, this result also indicates that non-expert ensembles should not be fed as is to deep ensemble AIs. Instead, careful preprocessing should be conducted to remove outliers.

Contour depths can enhance the analysis of individual cases, including information about the ensemble's variability. A noticeable example is the CTV1 (H&N). As Figure 7.1 shows, using the depth-trimmed ensemble leads to a substantive increase in agreement with the expert consensus for the STAPLE condition. We turn to the raw data to understand this phenomenon. The left pane of Figure 7.2 illustrates how the STAPLE of the full non-expert ensemble exhibits an elongated extension that is not present in the expert consensus, lowering the agreement between consensuses to $DSC = 0.85$. The STAPLE algorithm cannot remove this extension because several ensemble members include it. Using contour depth to trim outliers leads to a STAPLE consensus without the extension, increasing the consensuses' agreement to $DSC = 0.94$. Note that, as the fifty-percent confidence band on the figure's right pane indicates, most of the variability is still concentrated in the extension's region, which some members still include.

The results in this subsection show how the contour depth methodology can be used to enhance the robustness of downstream ensemble operations like consensus extraction. Further, depth scores permit analyzing the ensemble's variability by extracting uncertainty bands.

7.3.2. MULTI-MODAL ANALYSIS AND INTERACTIVE CANDIDATE SELECTION

We now turn to the usage of multi-modal contour depth for extracting representative shapes in the ensemble. Figure 7.3 illustrates the result of partitioning the non-expert ensemble of the Parotid_L (H&N) into two

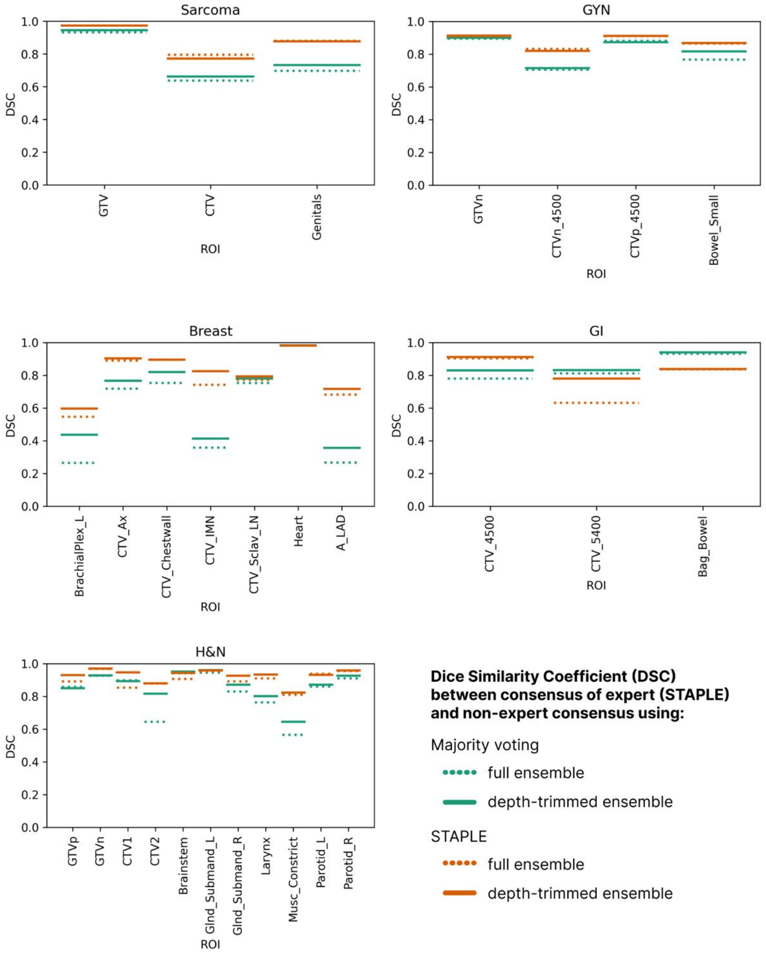


Figure 7.1.: DSC values comparing expert and non-expert consensus for five regions of interest in the C3RO challenge: sarcoma, breast, H&N, GYN, and GI cases. Expert consensus was extracted using the STAPLE algorithm in all cases. For the non-expert ensemble, consensus was derived using either majority voting (teal) or STAPLE (orange). Dotted lines represent consensus computed from the full ensemble, while solid lines indicate depth-trimmed ensembles, where only the top 70% of contours by contour depth were used.

modes of variation. The STAPLE consensuses extracted from these partitions are depicted using thicker colored lines.

Comparing the partitions' consensuses with the expert STAPLE

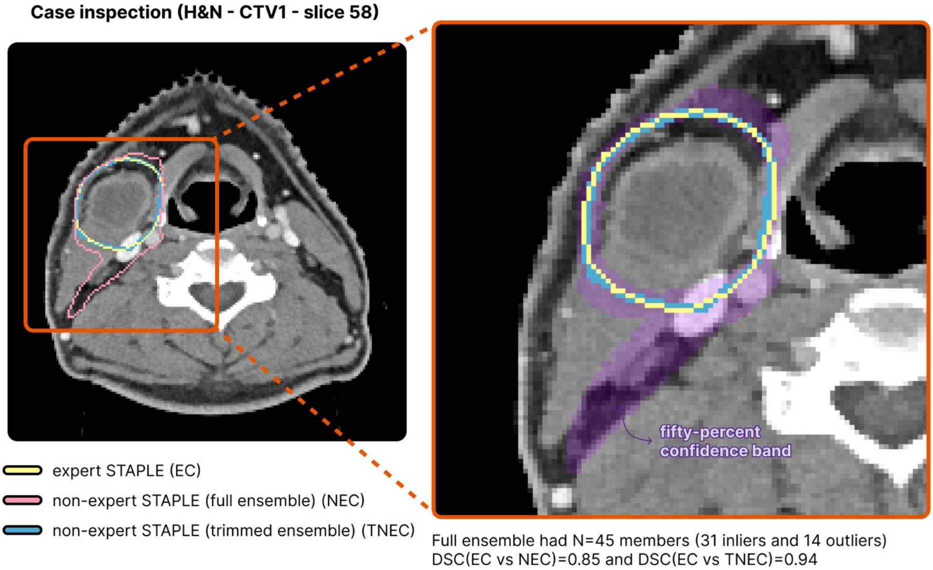

Case inspection (H&N - CTV1 - slice 58)
fifty-percent confidence band
🟨 expert STAPLE (EC)
🟥 non-expert STAPLE (full ensemble) (NEC)
🟦 non-expert STAPLE (trimmed ensemble) (TNEC)
Full ensemble had N=45 members (31 inliers and 14 outliers)
DSC(EC vs NEC)=0.85 and DSC(EC vs TNEC)=0.94

Figure 7.2.: Analysis of consensus extraction results for a slice of the CTV1 (H&N). The left pane shows the three STAPLE consensuses: expert (yellow) and non-expert, using the full (red) and depth-trimmed (blue) ensemble. The right pane focuses on the expert and depth-trimmed consensuses, indicating the variability of the depth-trimmed ensemble using a shaded purple region to denote the fifty-percent confidence band.

consensus (yellow), one sees that the orange contour underestimates the extend of the gland. In contrast, the teal shape includes the extension present in the expert STAPLE consensus, attaining a significantly higher $DSC = 0.94$. This scenario is representative of what can happen in clinical practice where clinicians are faced with segmentation ambiguities, which lead to variations in how they delineate. Multi-modal contour depth helps disentangling these trends, presenting a clearer overview of the ensemble.

The figure also hints at how multi-modal contour depth could power user-facing workflows. Specifically, allowing users to entertain and analyze several representative contours and eventually select the desired one could increase deep ensemble AIs' usability in clinical practice beyond robust consensus extraction. Further, clinicians could analyze the variability of the representative's partition using confidence bands before selecting a representative. This information can help guide the trimming of outliers. In the Parotid_L case, a clinician would choose the teal partition, which is closest to the expert consensus.

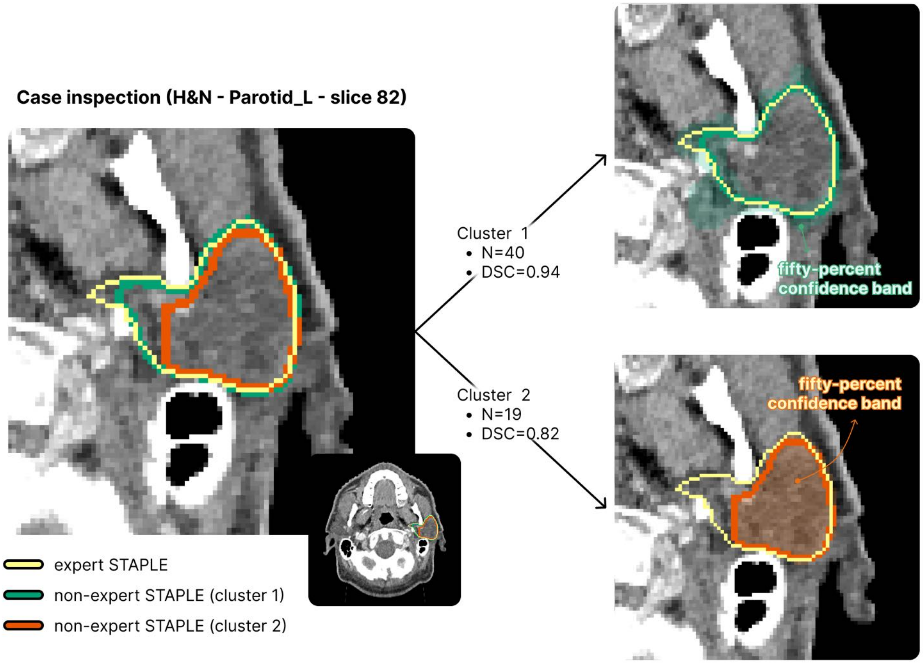

Case inspection (H&N - Parotid_L - slice 82)
expert STAPLE
non-expert STAPLE (cluster 1)
non-expert STAPLE (cluster 2)
Cluster 1
• N=40
• DSC=0.94
Cluster 2
• N=19
• DSC=0.82
fifty-percent confidence band
fifty-percent confidence band

Figure 7.3.: Multi-modal consensus extraction and analysis workflow for a slice of the Parotid_L (H&N). The left pane shows the STAPLE consensuses of the expert ensemble (yellow) and the two partitions of the non-expert ensemble (teal, orange) extracted using multi-modal contour depth. The right pane illustrates how a user can focus on one of the partitions, revealing a shaded region that corresponds to the variability of the top fifty (depth-wise) underlying ensemble members.

Figure 7.4 summarizes the results of the bi-modal contour depth experiment for the other ROIs and anatomical structures in the C3RO dataset. To reduce clutter, we focused on the consensus extracted using the STAPLE algorithm. Similarly to the previous subsection, each line denotes the DSC of a non-expert consensus versus the expert STAPLE one. The black dashed line corresponds to the non-expert STAPLE on the full dataset. The colored solid lines correspond to the two representative shapes.

As observed for most ROIs, the DSCs of the expert STAPLE and the STAPLE of the two representative shapes extracted by multi-modal contour depth exhibit a clear separation. This gap between DSC scores indicates that, in general, one of the variation modes is a better fit for the expert consensus. The wider gaps occur in H&N ROIs like the CTV2, the Larynx, and Parotid_L, which we visualized before. Interestingly, in many cases, the "winning" variation mode attains a DSC that improves the one obtained with the non-expert STAPLE consensus on the full dataset, which further demonstrates that trimming non-representative contours can lead to robustness gains.

Finally, the larger gaps between the DSCs of the representative shapes in Figure 7.4 are observed in target volumes. This outcome is expected, as these structures tend to have higher inter-observer variability due to their complexity and ambiguity. Similar to the Parotid_L example, Figure 7.5 illustrates the multi-modal analysis for CTV2 (H&N). As shown, the significantly lower agreement of the orange contour arises because it completely omits the right lobe. In contrast, the green contour aligns well with the expert consensus.

7.4. DISCUSSION

In this chapter, we comprehensively illustrated the application of the contour depth techniques developed in previous chapters in the radiotherapy setting. For this, we used as a case study the Contouring Collaborative for Consensus in Radiation Oncology (C3RO) challenge, which aimed to determine whether cheaper to obtain and more widely available non-expert segmentation ensembles can be used instead of expert ones for tasks like training deep ensemble AIs. Associated with the challenge is a dataset that offered expert and non-expert ensembles for ROIs across several anatomical regions.

We revisited the research question of the C3RO challenge using the depth methodology to re-analyze the results and found out that contour depth can help improve robustness. By trimming outliers from the non-expert ensemble before extracting the consensus, the agreement between expert and non-expert consensus increased across ROIs. An added benefit of the depths methodology is that it permits characterizing the variability of the trimmed ensemble using intuitive depth-sorted confidence bands. We leverage these bands to conduct uncertainty-aware inspection of specific cases, which helped better understand the nature of the disagreements between experts and non-experts.

In practice, ensembles often contain shapes with distinct characteristics due to data-gathering particularities such as different experience levels and varying visibility conditions. When using these ensembles to train deep ensemble AIs, AIs' predictions will likely also concentrate around a handful of representative shapes or modes of variation. We

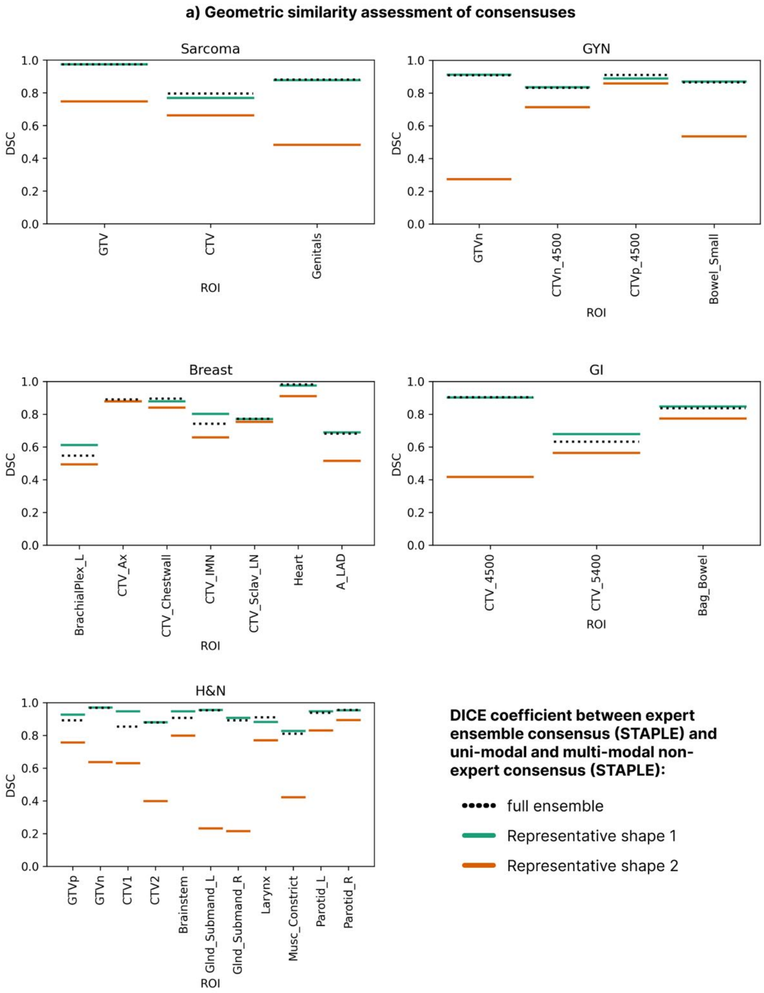


Figure 7.4.: DSC values comparing expert and several non-expert STAPLE consensus for five regions of interest in the C3RO challenge: sarcoma, breast, H&N, GYN, and GI cases. Black dotted lines represent consensus computed from the full ensemble, and colored solid lines indicate consensus extracted from a partition of the ensemble. Teal lines always correspond to the consensus that attained the largest agreement with the expert STAPLE.

showed how multi-modal contour depth can help disentangle these shapes before analysis, reducing bias due to the uni-modality assumption of traditional contour depth methods. Further, we investigated

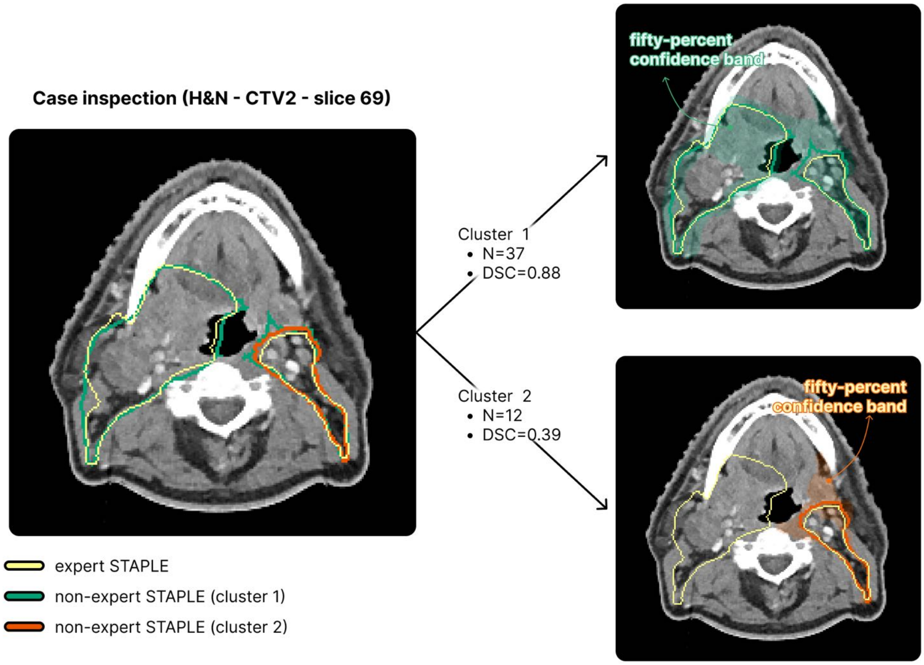

Case inspection (H&N - CTV2 - slice 69)
Cluster 1
• N=37
• DSC=0.88
Cluster 2
• N=12
• DSC=0.39
fifty-percent confidence band
fifty-percent confidence band
expert STAPLE
non-expert STAPLE (cluster 1)
non-expert STAPLE (cluster 2)

Figure 7.5.: Multi-modal consensus extraction and analysis workflow for a slice of the CTV2 (H&N). The left pane shows the STAPLE consensuses of the expert ensemble (yellow) and the two partitions of the non-expert ensemble (teal, orange) extracted using multi-modal contour depth. The right pane illustrates how a user can focus on one of the partitions, revealing a shaded region that corresponds to the variability of the top fifty (depth-wise) underlying ensemble members.

the viability of candidate selection workflows. Multi-modal contour depths could help extract representative shapes from deep ensemble AIs' predictions. Afterward, clinicians could select the most suitable candidate among the extracted ones, potentially reducing segmentation refinement effort and time.

This chapter serves as both a guide and a demonstration of contour depth functionality in the context of radiotherapy. Below, we outline the limitations of our analyses, which also suggest directions for future research.

Regarding the experiments conducted, a key limitation is the reliance on the untrimmed expert STAPLE consensus as the desired ground truth. A possible improvement would involve comparing ensembles directly

[7, 8] rather than using consensus methods as proxies. For extracting robust consensuses, another limitation is the use of a fixed proportion for outlier removal. A more adaptive approach, such as ensemble-specific thresholds based on depth scores, could provide better results [9]. Similarly, for visualizing uncertainty, we focused exclusively on the fifty-percent confidence bands. Allowing users to adjust the threshold for confidence band extraction could make the visualization more flexible, enabling tighter or wider bands tailored to specific applications.

Building on our findings, we identify three promising avenues for future research. First, extending the C3RO dataset to include more patients and regions of interest (ROIs) would strengthen the original challenge's conclusions and further validate the benefits of contour depth for increasing robustness and enabling candidate selection workflows. Second, exploring multi-modal contour depth beyond the bi-modal case is a compelling direction. This raises two questions: can more than two variation modes occur, and how should new workflows and visualization tools be designed to support users when multiple representatives are analyzed and selected? Finally, while contour depth operates at a global, shape-level scale, situations involving conflicting opinions may require more localized approaches [10]. Developing methods to address this need would be a step forward.

REFERENCES

[1] N. F. Chaves-de-Plaza, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "Inclusion Depth for Contour Ensembles". In: *IEEE Transactions on Visualization and Computer Graphics* 30.9 (2024), pp. 6560–6571. doi: 10.1109/TVCG.2024.3350076.

[2] N. F. Chaves-de-Plaza, M. Molenaar, P. Mody, M. Staring, R. v. Egmond, E. Eisemann, A. Vilanova, and K. Hildebrandt. "Depth for Multi-Modal Contour Ensembles". In: *Computer Graphics Forum* (2024). issn: 1467-8659. doi: 10.1111/cgf.15083.

[3] D. Lin, K. A. Wahid, B. E. Nelms, R. He, M. A. Naser, S. Duke, M. V. Sherer, J. P. Christodouleas, A. S. R. Mohamed, M. Cislo, J. D. Murphy, C. D. Fuller, and E. F. Gillespie. "E pluribus unum: prospective acceptability benchmarking from the Contouring Collaborative for Consensus in Radiation Oncology crowdsourced initiative for multiobserver segmentation". In: *Journal of Medical Imaging* 10.S1 (2023), S11903.

[4] M. V. Sherer, D. Lin, S. Elguindi, S. Duke, L.-T. Tan, J. Cacicedo, M. Dahele, and E. F. Gillespie. "Metrics to evaluate the performance of auto-segmentation for radiation treatment planning: A critical review". In: *Radiotherapy and Oncology* 160 (2021), pp. 185–191.

[5] K. J. Kiser, A. Barman, S. Stieb, C. D. Fuller, and L. Giancardo. "Novel Autosegmentation Spatial Similarity Metrics Capture the Time Required to Correct Segmentations Better Than Traditional Metrics in a Thoracic Cavity Segmentation Workflow". In: *Journal of Digital Imaging* 34.3 (2021), pp. 541–553.

[6] S. Warfield, K. Zou, and W. Wells. "Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation". In: *IEEE Transactions on Medical Imaging* 23.7 (2004), pp. 903–921. doi: 10.1109/TMI.2004.828354.

[7] S. Kohl, B. Romera-Paredes, C. Meyer, J. De Fauw, J. R. Ledsam, K. Maier-Hein, S. M. A. Eslami, D. Jimenez Rezende, and O. Ronneberger. "A Probabilistic U-Net for Segmentation of Ambiguous Images". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc., 2018.

[8] M. Monteiro, L. Le Folgoc, D. Coelho de Castro, N. Pawlowski, B. Marques, K. Kamnitsas, M. van der Wilk, and B. Glocker. "Stochastic Segmentation Networks: Modelling Spatially Correlated Aleatoric Uncertainty". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 12756–12767.

[9] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. "Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles". In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2713–2722. doi: 10.1109/TVCG.2013.143.

[10] M. Mirzargar and R. T. Whitaker. "Representative Consensus from Limited-Size Ensembles". In: *Computer Graphics Forum* 37.3 (2018), pp. 13–22.

8

CONCLUSION

Deep ensemble Artificial Intelligence (AI) methods are an exciting research avenue in medical image segmentation. While standard auto-segmentation AIs output one segmentation, deep ensemble AIs attempt to model clinicians' behavior, yielding a set of plausible candidates. Compared to ensembles manually created by clinicians, deep ensemble AIs operate at a fraction of the time and cost. These efficiency gains and the additional information provided by segmentation ensembles have the potential to transform and streamline the AI-supported segmentation workflow, unlocking new use cases and opportunities for enhanced clinician-AI collaboration. In turn, this will reduce the resource footprint of adaptive radiotherapy (RT) workflows, paving the way for its clinical implementation.

Deep ensembles can already be used to improve segmentation quality and robustness [1], and to guide the development of more performant auto-segmentation AIs [2]. Nevertheless, realizing their full potential for supporting the clinician-centered quality assessment process requires developments on multiple fronts.

The first part of this dissertation contributes a better understanding of the clinician-driven quality assessment process and the scenarios in which deep ensembles can provide value. Chapter 2 uncovered and characterized the central tasks in quality assessment process: navigation, error detection and correction. Chapter 3 evaluated delineation error detection technology (DEDS). By foregoing slices with no segmentation failures, we found DEDS can help clinicians more quickly assess contours' quality. More generally, these chapters revealed the visual ensemble analysis process as a foundational piece of the quality assessment process, guiding and potentially supporting clinicians' actions.

In the second part of this dissertation, we focused obstacles for effective and efficient visual ensemble analysis. Due to the large number of ensemble members and their multi-dimensional nature (spawning multiple spatial and even temporal dimensions), existing visual analysis methods can take too long to compute, fatigue clinicians, and be prone to clutter, obscuring critical patterns.

The Inclusion Depth (ID) and multi-modal ID methodologies introduced in Chapters 4 and 5 allow the detection of the main trends in the ensemble and the extraction of statistics that, when paired with contour boxplots, permit clinicians to get an overview of the ensemble in seconds. We showed how the developed contour depth methodologies can be applied to practical scenarios in RT in Chapter 7. Finally, Chapter 6 tackles the challenge of partitioning the ensemble spatial domain which could be useful to localize or accelerate analyses. For this, we presented the Local-to-Global Correlation Clustering (LoGCC) framework, which permits rapidly partitioning the ensemble spatial domain into regions where ensemble members' behavior is consistent.

The proposed techniques pave the way to leveraging deep ensembles to support clinicians-driven segmentation quality assessment tasks. In the following, we sketch this vision and discuss potential implementation challenges that represent future work opportunities.

Navigation and Error Detection DEDS point clinicians to relevant slices of the patient's images and segmentations. Although anomalous slice detection already provides structure to investigate AI-generated segmentations more efficiently, we found that slice-based navigation can be cumbersome and fatiguing as the system guides clinicians to spatially disconnected volume regions, breaking their mental frame. An interesting research avenue is to use a navigational unit that is more semantically meaningful to the quality assessment process. For instance, three-dimensional regions containing potential errors could be presented to the clinician. How to extract these regions and adequately visualize them are questions that remain open.

Another interesting result is that infusing DEDS with a notion of priority like clinical significance can boost the time gains offered. We found that coupling error detection with a dose-based metric, which permits clinicians to decide whether segmentation failures can affect the patient's treatment, leads to fewer editing and, consequently, reduced quality assessment times. Although some work has been done in providing dose estimates [3], dose information might not always be available in practice. Further, depending on the application domain, the administered dose might not be a signal of error significance/priority. Therefore, we believe developing these priority metrics jointly with relevant domain experts (e.g., clinicians in radiotherapy) is a promising future direction for DEDS development.

Error Analysis and Editing We developed and published the contour depth Python package, which enables using ID and multi-modal ID as a computational building blocks. Similarly, LoGCC will soon also be available as a Python package. Due to their modular nature, the developed ID and LoGCC packages can be used as components within a larger system or workflow.

Within the frame of this dissertation, an exciting direction is integrating multi-modal ID and LoGCC into a more efficient ensemble-supported quality assessment process that delays expensive editing interactions like manual brushing or scribbling. In the first step, the clinician picks one of the variation modes yielded by the multi-modal ID as the global representative. If further localized editing is needed, the clinician can fix the portion of the segmentation they are satisfied with and focus on a region of interest that requires editing. Within this region, they can modify the selected global representative by, for instance, adding or subtracting highly consistent regions extracted by LoGCC. The process repeats until the clinician is satisfied with the segmentation or until they find an anomaly whose solution is not contained in the ensemble, prompting the usage of traditional segment editing tools.

The successful implementation of the ensemble-supported quality assessment process requires careful consideration of what information to present to clinicians and how to present it. It is unclear whether variation modes or LoGCC outputs should be shown to clinicians, especially given the time pressures and fatigue they often face. For example, introducing new uncertainty channels, such as confidence bands, might exacerbate cognitive load and hinder usability [4]. Alternatively, uncertainty information could be displayed on demand, triggered by clicks or other interactions [5]. To address these challenges, a clinician-centered validation of uncertainty-aware segmentation visualizations is essential to identify the most effective approaches for ensemble assessment and candidate selection tasks. Finally, another potential issue is that the local selection process could produce consensus segmentations with jagged lines, especially when combining distinct variation modes. A post-processing step that ensures valid and clinically acceptable segmentation masks is thus necessary [6].

Beyond the technical challenges that will inevitably arise when implementing workflows like those described before, we identified two general issues central to realizing the vision of an ensemble-supported quality assessment process: modeling quality of deep ensembles and attitudes of domain experts and potential users towards the deep ensemble AI technology. Below, we discuss these issues and the future work avenues they span. We finalize by discussing how the findings and methods of this dissertation apply to other scenarios and domains.

Quality of deep ensemble modeling The proposed usage of deep ensembles to support the segmentation QA process rests on the assumption that they can accurately model clinicians' behavior, producing numerous, diverse, and plausible segmentation hypotheses necessary to reliably obtain distinct variation modes, and extract their consensus and confidence bands. Deep ensemble methods often struggle to model clinicians' behavior faithfully. Deep ensembles can contain unrealistic-looking candidates or candidates that are not diverse enough and exhibit variability in areas where clinicians would not due to the entanglement of different uncertainty sources [7]. Deep ensemble modeling is a very active research area with significant investment in model improvements [8–12] and the creation of novel and larger multi-annotator datasets [13–15]. Therefore, we expect that significant progress will follow in the coming years.

Attitudes toward deep ensemble AI Perhaps the greatest challenge to using ensemble-based technology in clinical practice is clinicians' perception. The prospect of wrong decisions due to a misleading representation of uncertainty and the lack of complete understanding of their effect on patient outcomes can negatively impact clinicians' adoption of ensembles and their affordances (e.g., variation modes, uncertainty, and partitioning into consistent regions) [16]. Nevertheless, it is worth noting that ensemble-based features like candidate selection do not require perfect accuracy to be useful. Similar to other technologies that face challenges in clinical adoption, such as deep learning, clinicians' attitudes and trust can be positively influenced through several approaches. For instance, involving clinicians in the development and refinement of ensemble-based methods or using retrospective datasets in a low-stakes research environment to demonstrate to clinicians the positive impact of these methods before moving into clinical practice [17].

Application to other domains To conclude, we stress the generality of the ensemble-based methods proposed in this dissertation. We designed and implemented them with modularity and reusability in mind. Therefore, we expect the proposed methods to be applicable to other scenarios and to be able to support novel workflows in other domains. In radiotherapy, DEDS need not be limited to model deployment. DEDS are commonly used in the model development stage to permit developers to find areas where the model needs more annotation or improvement [2]. ID and LoGCC permit analysis and communication of results of IOV studies, which are crucial to determine margins and interventions to improve segmentation quality. Beyond the medical domain, our methods can be used in any field where ensembles of segmentations or even scalar fields (from which iso-contours can be extracted) play a role. Examples of other possible application domains are cell biology, where one is interested in analyzing cells' morphology [18], and meteorology where our methods permit extracting the main trends in meteorological forecasts and determining agreement areas between ensemble members.

REFERENCES

[1] M. Wolf, J. Krause, P. A. Carney, A. Bogart, and R. H. J. M. Kurvers. “Collective Intelligence Meets Medical Decision-Making: The Collective Outperforms the Best Radiologist”. In: *PLOS ONE* 10.8 (Aug. 2015), pp. 1–10.

[2] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Z. Chen. “Suggestive Annotation: A Deep Active Learning Framework for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2017*. Ed. by M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, and S. Duchesne. Cham: Springer International Publishing, 2017, pp. 399–407.

[3] B. Roberfroid, J. A. Lee, X. Geets, E. Sterpin, and A. M. Barragán-Montero. “DIVE-ART: A tool to guide clinicians towards dosimetrically informed volume editions of automatically segmented volumes in adaptive radiation therapy”. In: *Radiotherapy and Oncology* 192 (2024), p. 110108.

[4] M. Huet-Dastarac, N. van Acht, F. Maruccio, J. van Aalst, J. van Oorschodt, F. Cnossen, T. Janssen, C. Brouwer, A. Barragan Montero, and C. Hurkmans. “Quantifying and visualising uncertainty in deep learning-based segmentation for radiation therapy treatment planning: What do radiation oncologists and therapists want?” In: *Radiotherapy and Oncology* 201 (2024), p. 110545.

[5] B. Shneiderman. “The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations”. In: *The Craft of Information Visualization*. Ed. by B. B. BEDERSON and B. SHNEIDERMAN. Interactive Technologies. San Francisco: Morgan Kaufmann, 2003, pp. 364–371.

[6] M. Mirzargar and R. T. Whitaker. “Representative Consensus from Limited-Size Ensembles”. In: *Computer Graphics Forum* 37.3 (2018), pp. 13–22.

[7] A. Kendall and Y. Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017.

[8] S. A. A. Kohl, B. Romera-Paredes, C. Meyer, J. D. Fauw, J. R. Ledsam, K. H. Maier-Hein, S. M. A. Eslami, D. J. Rezende, and O. Ronneberger. “A probabilistic U-net for segmentation of ambiguous images”. In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS’18. Montréal, Canada: Curran Associates Inc., 2018, pp. 6965–6975.

[9] S. A. A. Kohl, B. Romera-Paredes, K. Maier-Hein, D. J. Rezende, S. M. A. Eslami, P. Kohli, A. Zisserman, and O. Ronneberger. “A Hierarchical Probabilistic U-Net for Modeling Multi-Scale Ambiguities”. In: *ArXiv* abs/1905.13077 (2019). url: https://api.semanticscholar.org/CorpusID:170079074.

[10] M. Monteiro, L. L. Folgoc, D. C. de Castro, N. Pawlowski, B. Marques, K. Kamnitsas, M. van der Wilk, and B. Glocker. “Stochastic segmentation networks: modelling spatially correlated aleatoric uncertainty”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS ’20. Vancouver, BC, Canada: Curran Associates Inc., 2020. isbn: 9781713829546.

[11] M.-I. Georgescu, R. T. Ionescu, and A. I. Miron. “Diversity-Promoting Ensemble for Medical Image Segmentation”. In: *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*. SAC ’23. Tallinn, Estonia: Association for Computing Machinery, 2023, pp. 599–606. isbn: 9781450395175. doi: 10.1145/3555776.3577682. url: https://doi.org/10.1145/3555776.3577682.

[12] M. Ng, F. Guo, L. Biswas, S. E. Petersen, S. K. Piechnik, S. Neubauer, and G. Wright. “Estimating Uncertainty in Neural Networks for Cardiac MRI Segmentation: A Benchmark Study”. In: *IEEE Transactions on Biomedical Engineering* 70.6 (2023), pp. 1955–1966. doi: 10.1109/TBME.2022.3232730.

[13] S. Nikolov, S. Blackwell, A. Zverovitch, R. Mendes, M. Livne, J. De Fauw, Y. Patel, C. Meyer, H. Askham, B. Romera-Paredes, C. Kelly, A. Karthikesalingam, C. Chu, D. Carnell, C. Boon, D. D'Souza, S. A. Moinuddin, B. Garie, Y. McQuinlan, S. Ireland, K. Hampton, K. Fuller, H. Montgomery, G. Rees, M. Suleyman, T. Back, C. O. Hughes, J. R. Ledsam, and O. Ronneberger. "Clinically Applicable Segmentation of Head and Neck Anatomy for Radiotherapy: Deep Learning Algorithm Development and Validation Study". In: *J Med Internet Res* 23.7 (July 2021), e26151.

[14] S. G. Armato III, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, B. Zhao, D. R. Aberle, C. I. Henschke, E. A. Hoffman, E. A. Kazerooni, H. MacMahon, E. J. R. van Beek, D. Yankelevitz, A. M. Biancardi, P. H. Bland, M. S. Brown, R. M. Engelmann, G. E. Laderach, D. Max, R. C. Pais, D. P.-Y. Qing, R. Y. Roberts, A. R. Smith, A. Starkey, P. Batra, P. Caligiuri, A. Farooqi, G. W. Gladish, C. M. Jude, R. F. Munden, I. Petkovska, L. E. Quint, L. H. Schwartz, B. Sundaram, L. E. Dodd, C. Fenimore, D. Gur, N. Petrick, J. Freymann, J. Kirby, B. Hughes, A. Vande Casteele, S. Gupte, M. Sallam, M. D. Heath, M. H. Kuhn, E. Dharaiya, R. Burns, D. S. Fryd, M. Salganicoff, V. Anand, U. Shreter, S. Vastagh, B. Y. Croft, and L. P. Clarke. "The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans". In: *Medical Physics* 38.2 (2011), pp. 915–931.

[15] D. Lin, K. A. Wahid, B. E. Nelms, R. He, M. A. Naser, S. Duke, M. V. Sherer, J. P. Christodouleas, A. S. R. Mohamed, M. Cislo, J. D. Murphy, C. D. Fuller, and E. F. Gillespie. "E pluribus unum: prospective acceptability benchmarking from the Contouring Collaborative for Consensus in Radiation Oncology crowdsourced initiative for multiobserver segmentation". In: *Journal of Medical Imaging* 10.S1 (2023), S11903.

[16] C. Gillmann, N. N. Smit, E. Gröller, B. Preim, A. Vilanova, and T. Wischgoll. "Ten Open Challenges in Medical Visualization". In: *IEEE Computer Graphics and Applications* 41.5 (2021), pp. 7–15. doi: 10.1109/MCG.2021.3094858.

[17] O. Asan, A. E. Bayrak, and A. Choudhury. "Artificial Intelligence and Human Trust in Healthcare: Focus on Clinicians". In: *J Med Internet Res* 22.6 (June 2020), e15154.

[18] A. Myers and N. Miolane. "Regression-Based Elastic Metric Learning on Shape Spaces of Cell Curves". In: *NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*. 2022. url: https://openreview.net/forum?id=8YKd0rwc4mu.

A

IMPLEMENTATION OF DELINEATION ERROR DETECTION SYSTEMS IN TIME-CRITICAL RADIOTHERAPY: DO AI-SUPPORTED OPTIMIZATION AND HUMAN PREFERENCES MEET?

DEDS DEVELOPMENT

In this section, we outline the development of our Delineation Error Detection System (DEDS) used in the workflow comparison user study (Sec. 3.5). We engaged in a co-development process with RO1 (RO from Utrecht UMC) and RO2 (RO from Leiden University Medical Center), involving multiple sessions where they used the tool for error detection and participated in structured discussions regarding tool usability and information source suitability. Our analysis involved logging clinicians' interactions and transcribing discussions, with relevant excerpts provided below.

CLINICAL DELINEATION SOFTWARE

Fig. 3.3's top panel displays a standard open-source delineation software's graphical user interface (GUI), consisting of two primary sections: the slice explorer (light blue rectangle) listing anatomical structures for delineation and the slice viewer (orange rectangle) for navigating 3D images via scrolling or navigation keys, supporting zooming and panning, and enabling pixel editing using tools like brushes or polygon pens. Our custom implementation, based on this GUI, was developed to support the slice-based error detection task. While we initially considered using existing delineation software, their closed source code or complexity hindered our envisioned extensions. Therefore, we re-implemented essential functionalities, excluding editing features, and instead used key presses to indicate editing intentions, as described in the subsequent section on extending the prototype.

ERROR DETECTION AND PRIORITIZATION VIA PER-SLICE SCORES

The bottom panel of Fig. 3.3 shows the GUI of the DEDS prototype. Similar to delineation software it has a slice explorer and viewer. Nevertheless, we extended the slice explorer with two features that permit slice-driven error detection. First, the list offers a higher slice-level granularity level. Traditional software only allows browsing a list of OARS. The DEDS slice explorer permits drilling down the OAR into the slices that it spans. Furthermore, it permits sorting each OAR's slices based on user-defined scores as defined in Sec. 3.3.3. The bottom left area of the slice explorer in Fig. 3.3 shows the score definition widget.

CLINICIANS' FEEDBACK

The DEDS prototype underwent significant changes based on feedback from RO1 and RO2, including the addition of contextual information and image overlay features, customization of color maps, and simplification of score displays. Clinicians' feedback influenced workflow improvements, such as grouping slices by structure in the slice explorer for a less overwhelming experience. Initial impressions of `unc` and `error` were mixed, with clinicians finding them limited and potentially misleading, leading to reduced trust in the system. To address this, explanations were provided during the workflow comparison study. In contrast, clinicians reacted positively to `dose` information, suggesting predefined settings per organ, with an emphasis on maximum dose and gradient magnitude (`grad_dose`) as valuable additions to the information sources. These enhancements aimed to enhance DEDS usability and effectiveness.

B

INCLUSION DEPTH FOR CONTOUR ENSEMBLES

PROPERTIES OF ID

In this appendix, we provide sketches of proofs of the theoretical properties of ID listed in Sec. IV. We start by showing that ID is invariant to certain transformations of the domain.

Proposition 1: ID's results are invariant to homeomorphic transformations of the domain.
Proof sketch: Let $\phi : \Omega \rightarrow \Psi$ be a homeomorphism. Due to the bijectivity of ϕ, for all i, j, $in(c_i) \subset in(c_j)$ holds if and only if $\phi(in(c_i)) \subset \phi(in(c_j))$. Therefore, the quantities IN_{in} and IN_{out} defined in Eqs. 6 and 7 are equal. It follows that the ID is also the same. We consider continuous transformations because we represent the contours as level-sets of continuous functions.

We now proceed to establish the relationship between Half-Region Depth (HRD) and ID.

Proposition 2: In the special case in which ID is applied to graphs of functions, it is equivalent to HRD.
Proof sketch: Let $\{x_1, x_2, \ldots, x_N\} \in X$ be an ensemble of functions in a region $U = I \times \mathbb{R}$ with graph $G(x) = \{(t, x(t)) : t \in I\}$. Computing the HDR of a function x_i entails counting the number of times $G(x_i) \subset epi(x_j)$ and $G(x_i) \subset hypo(x_j)$ for all $x_j \in X$, dividing by N and selecting the minimum of the two quantities. The same result can be obtained evaluating $hypo(x_i) \subset hypo(x_j)$ and $hypo(x_j) \subset hypo(x_i)$ instead. Establishing the correspondence between *hypo* and *in* in Eq. 2 makes evident how this formulation matches the inside/outside relationships in Equations 6 and 7.

Proposition 3: If an ensemble of contours is produced by a homeomorphism (bijective, bi-continuous transformation) applied to an ensemble of function graphs, computing the contours' ID is equivalent to computing the functions' HRD.
Proof sketch: This follows by combining Propositions 1 and 2.

The following propositions establish additional properties of ID.

Proposition 4: ID is invariant with respect to the definition of the contours' inside regions.
Proof sketch: We start with an analysis of contours and their masks obtained by thresholding a field $F_i(x, y) = q$. Computing ID requires calculating the contours' inside regions as $in(c_i) = \{p \in \Omega | F_i(p) < q\}$. If we perform an analysis on the field $F_i'(x, y) = -F_i(x, y)$ then the inside and outside regions of the contour flip. For $c_i, c_j \in C$, if in the original formulation $in(c_i) \subset in(c_j)$, with the updated formulation it holds that $in'(c_j) \subset in'(c_i)$. It is straightforward to see how, with this change, the definitions of $\text{IN}_{in}(c_i)$ and $\text{IN}_{out}(c_i)$ in Eqs. 6 and 7 flips. Given that the ID is the minimum of the two quantities, it remains the same under the sign change of F and q.

Finally, we also show that Epsilon Inclusion Depth (eID) is also invariant to certain transformations.

Proposition 6: eID's results are invariant to area-preserving transformations of the domain.
Proof sketch: Let $\phi : \Omega \to \Psi$ be an area-preserving bijection (equiareal or authalic map). Due to the area-preserving property, the numerator and denominator in Eq. 9 remain constant. Therefore, the quantities $\text{IN}_{in}^{\epsilon}$ and $\text{IN}_{out}^{\epsilon}$ defined in Eq. 10 are equal. It follows that the eID is also the same.

COMPARISON OF ID AND CBD MEDIANS AND α-TRIMMED MEANS

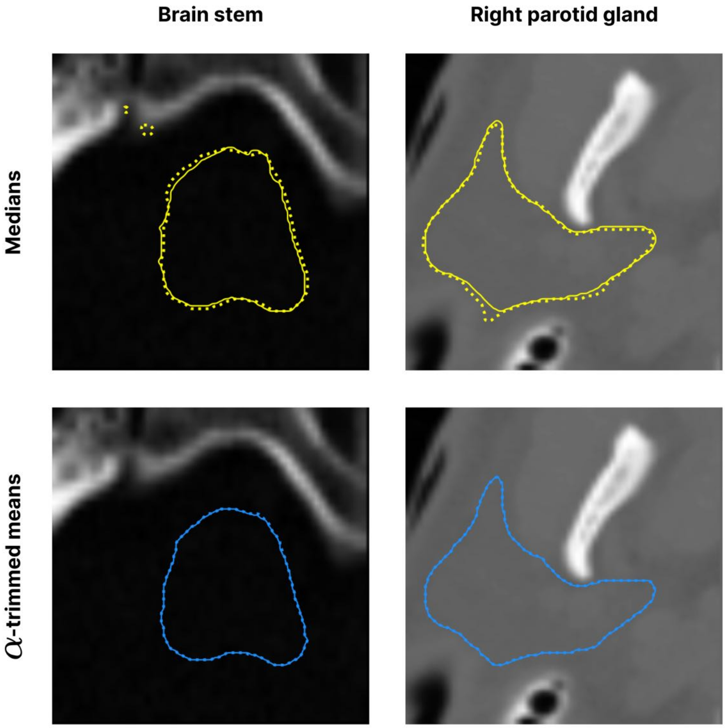
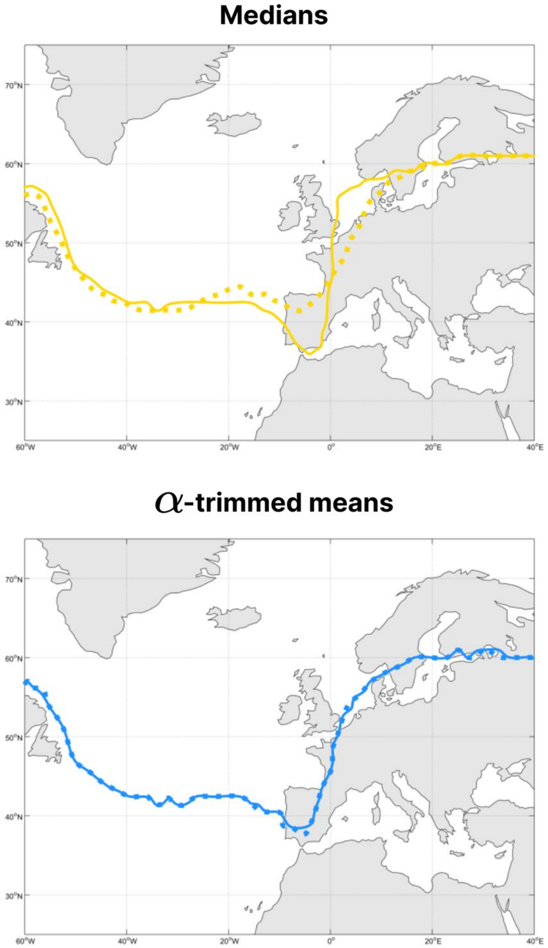


Figure B.1.: Comparison of medians (yellow) and α-trimmed means (blue) that CBD (solid lines) and ID (dotted lines) yield for the brain stem (left column) and right parotid gland (right column) of a head-and-neck cancer patient. The median is the contour with the highest CBD or ID depth. For the α-trimmed mean, we used $\alpha = 0.1$.



Medians

α-trimmed means

Figure B.2.: Comparison of medians (yellow) and α-trimmed means (blue) that CBD (solid lines) and ID (dotted lines) yield for the ensemble of 500 hPa geopotential height contour lines. The median is the contour with the highest CBD or ID depth. For the α-trimmed mean, we used $\alpha = 0.2$.

C

LOGCC: LOCAL-TO-GLOBAL CORRELATION CLUSTERING FOR SCALAR FIELD ENSEMBLES

LEMMAS CONCERNING CORRELATION TRANSITIVITY

In this section, we prove two lemmas concerned with weak correlation transitivity of Pearson correlation. For two vectors $S_i, S_j \in \mathbb{R}^N$ Pearson's correlation is given by

$$\rho(S_i, S_j) = \frac{\sum_{n=1}^{N}(S_i(n) - \bar{S}_i)(S_j(n) - \bar{S}_j)}{\sqrt{\sum_{n=1}^{N}(S_i(n) - \bar{S}_i)^2 \sum_{n=1}^{N}(S_j(n) - \bar{S}_j)^2}}, \tag{C.1}$$

where $S_i(n)$ denotes the entries of S_i and $\bar{S}_i$ the mean of the N entries of S_i. The basis of the proofs is a geometric interpretation of the Pearson correlation, which is also discussed in [1]. We consider the centred an normalized vectors s_i given by

$$s_i = \frac{S_i - \bar{S}_i e}{\left\|S_i - \bar{S}_i e\right\|}, \tag{C.2}$$

where $\|\|$ is the norm in $\mathbb{R}^N$ and $e \in \mathbb{R}^N$ the vector with all entries equal to one. Note, that the factor for normalizing the vectors is the reciprocal of the standard deviation of S_i. Then, the Pearson correlation of S_i and S_j is the scalar product of the vectors s_i and s_i

$$\rho(S_i, S_j) = \left\langle s_i, s_j \right\rangle. \tag{C.3}$$

This implies that the Pearson correlation of S_i and S_j equals the cosine of the angle ϕ_{ij} between the unit vectors s_i and s_j. Fig C.1 depicts the relationships between unit vectors and the angles between them discussed in the two lemmas below.

Lemma 3 *Consider a triplet $S_i, S_j, S_k \in \mathbb{R}^N$ and assume $\rho(S_i, S_j) \geq \rho$ and $\rho(S_i, S_k) \geq \rho$ for some $\rho \in [0,1]$. Then,*

$$\rho(S_j, S_k) \geq cos(2arccos(\rho)). \tag{C.4}$$

Proof. For S_i, S_j, S_k, we denote by s_i, s_j, s_k the centered an normalized vectors as in Eq. C.2. Denoting the angle between the unit vectors s_i and s_j by ϕ_{ij}, Eq. C.3 implies

$$\langle s_i, s_j \rangle = \cos(\phi_{ij}). \tag{C.5}$$

The assumptions $\rho(S_i, S_j) \geq \rho$ and $\rho(S_i, S_k) \geq \rho$ imply

$$\phi_{ij} \leq \arccos(\rho) \quad \text{and} \quad \phi_{ik} \leq \arccos(\rho). \tag{C.6}$$

By our assumption that $\rho \in [0,1]$, ϕ_{ij} and ϕ_{ik} are smaller or equal to $\frac{\pi}{2}$. The triangle inequality for angles implies that the angle ϕ_{jk} has to be smaller or equal to the sum of the angles ϕ_{ij} and ϕ_{ik}. Hence

$$\phi_{jk} \leq 2\arccos(\rho). \tag{C.7}$$

Since the cosine function is monotonically decreasing on the interval $[0,\pi]$, Eqs. C.7 implies

$$\cos(\phi_{jk}) \geq \cos(2\arccos(\rho)). \tag{C.8}$$

Finally, we combine Eqs. C.8, C.5 and C.3 to get Eq. C.4. ■

Lemma 4 *Consider a quadruple $S_i, S_j, S_k, S_l \in \mathbb{R}^N$ and assume $\rho(S_i, S_j) \geq \rho_g$, $\rho(S_i, S_k) \geq \rho_l$ and $\rho(S_j, S_l) \geq \rho_l$ for some $\rho_l, \rho_g \in [0.5,1]$. Then,*

$$\rho(S_l, S_k) \geq cos(2arccos(\rho_l) + arccos(\rho_g)). \tag{C.9}$$

Proof. For S_i, S_j, S_k, S_l, we consider the centered an normalized vectors s_i, s_j, s_k, s_j as in Eq. C.2 and denote by ϕ_{ij} the angle between s_i and s_j. Using Eqs. C.3 and C.5, the assumptions $\rho(S_i, S_j) \geq \rho_g$, $\rho(S_i, S_k) \geq \rho_l$ and $\rho(S_j, S_l) \geq \rho_l$ imply

$$\phi_{ij} \leq \arccos(\rho_g) \quad \text{and} \quad \phi_{ik}, \phi_{jl} \leq \arccos(\rho_l). \tag{C.10}$$

Since $\rho_g, \rho_l \in [0.5,1]$, ϕ_{ij}, ϕ_{ik} and ϕ_{jl} are smaller or equal to $\frac{\pi}{3}$. Then, the triangle inequality for angles yields

$$\phi_{kl} \leq \phi_{ik} + \phi_{ij} + \phi_{jl},$$

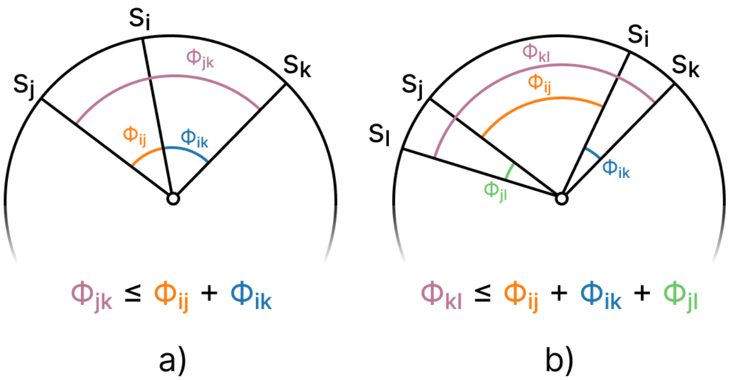


Figure C.1.: Geometric perspective of relationships between correlations in a) triples and b) quadruples of cells in the ensemble. Angles inversely relate to correlations between vectors via $\rho = \cos(\phi)$. The scheme represents the situations that arise when building clusters in the a) local and b) global steps of LoGCC. For simplicity, relationships are presented using a circle. In higher dimensions, angles get smaller, but the relationships hold.

which combined with (C.10) implies

$$\phi_{kl} \leq 2\arccos(\rho_l) + \arccos(\rho_g). \tag{C.11}$$

Since the cosine function is monotonically decreasing on the interval $[0, \pi]$, Eq. C.11 implies

$$\cos(\phi_{kl}) \geq \cos(2\arccos(\rho_l) + \arccos(\rho_g)). \tag{C.12}$$

Finally, combining Eqs. C.12, C.5 and C.3 yields Eq. C.9. ∎

PIVOT-BASED CC ALGORITHMS

Algorithms 6 and 7 describe the Pivot and CN-Pivot correlation clustering (CC) methods applied to a scalar field S with a correlation threshold ρ_t. The first step in both algorithms is to convert the scalar field into a graph, where the cells correspond to vertices, and edges connect all vertex pairs. The edge set is separated into positive and negative subsets based on the correlation between the connected cells.

Pivot builds clusters by randomly selecting a pivot and including all vertices sharing a positive edge with it. This process repeats until all vertices have a cluster. CN-Pivot starts with a pre-processing step where correlation neighborhoods $\eta(V_i)$ are computed for all vertices.

The algorithm then sorts the vertices in decreasing order of correlation neighborhood size. A correlation neighborhood forms a new cluster in the main loop if none of its correlation neighbors are already members of a cluster. Note that CN-Pivot may leave some vertices unassigned.

Algorithm 6 Pivot

```
Require: $S, \rho_t$                    ▷ Scalar field ensemble and correlation threshold
  $G^{\rho_t} = (V, E^+, E^-)$                           ▷ Define +/- edges using $\rho_t$
  $\pi \leftarrow$ uniform random permutation of $V$
  $\Psi = \emptyset; I = \emptyset$                                  ▷ Pivots and clusters
  while $\pi \neq \emptyset$ do
    $V_\psi \leftarrow pop(\pi, 1)$                              ▷ Pop first vertex in $\pi$
    $\Psi \leftarrow \Psi \cup V_\psi; I_\psi \leftarrow \{V_\psi\}$                        ▷ Initialize cluster
    for $V_\pi \in \pi$ do
      if $(V_\psi, V_\pi) \in E^+$ then
        $I_\psi \leftarrow I_\psi \cup V_\pi$
        $\pi \leftarrow \pi - \{V_\pi\}$
      end if
    end for
    $I \leftarrow I \cup I_\psi$
  end while
  return $\Psi, I$
```

Algorithm 7 CN-Pivot

```
Require: $S, \rho_t$                    ▷ Scalar field ensemble and correlation threshold
  $G^{\rho_t} = (V, E^+, E^-)$                           ▷ Define +/- edges using $\rho_t$
  $\eta^{\rho_t}(V_i) = \{V_j \in V | E_{ij} \in E^+\}$                  ▷ Correlation neighborhoods
  $\pi \leftarrow$ permutation of $V$ in decreasing order of correlation neighborhood
  cardinality
  $\Psi = \emptyset; I = \emptyset$                                  ▷ Pivots and clusters
  while $\pi \neq \emptyset$ do
    $V_\psi \leftarrow pop(\pi, 1)$                              ▷ Pop first vertex in $\pi$
    if $\eta^{\rho_t}(V_\psi) \cap \eta^{\rho_t}(V'_\psi) = \emptyset \forall V'_\psi \in \Psi$ then
      $\Psi \leftarrow \Psi \cup V_\psi; I_\psi \leftarrow \eta^{\rho_t}(V_\psi)$                    ▷ Build cluster
    end if
  end while
  return $\Psi, I$
```

DETAILS SYNTHGRID DATASET

The SynthGrid dataset permits evaluating the scaling behavior of CC methods by modulating different parameters like the field size, the number of clusters, and the proportion of unique clusters, which leads to clusters with disconnected components. Figure C.2 presents examples of the ensemble members (a) and of the CC obtained when changing the field size (b) and the number of clusters (c).

ADDITIONAL EXAMPLE CORRELATION THRESHOLD EXPLORATION

Fig. C.3 presents another example of the exploratory workflow for correlation thresholds applied to the HaN-Brainstem dataset. In this example, the borders of clusters near the target segmentation boundary do not align perfectly with it. This misalignment could indicate uncertainty about the boundary's exact location, which is understandable given the insufficient contrast information in the CT scan. As a result, the deep neural network may be inferring the brainstem's shape based on segmentations previously during training rather than on the image data.

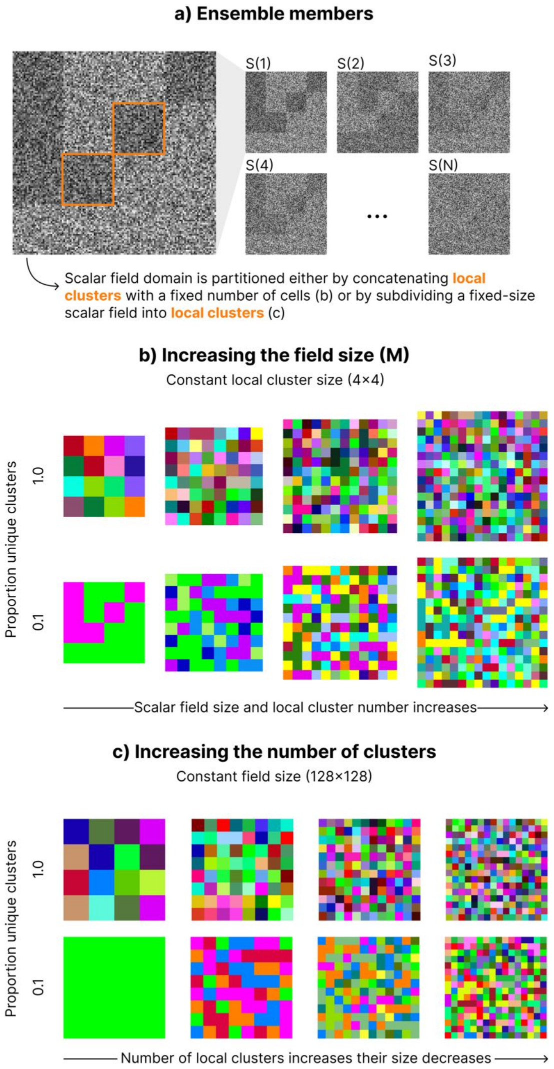

a) Ensemble members
S(1)
S(2)
S(3)
S(4)
...
S(N)
Scalar field domain is partitioned either by concatenating local clusters with a fixed number of cells (b) or by subdividing a fixed-size scalar field into local clusters (c)
b) Increasing the field size (M)
Constant local cluster size (4×4)
Proportion unique clusters
1.0
0.1
Scalar field size and local cluster number increases
c) Increasing the number of clusters
Constant field size (128×128)
Proportion unique clusters
1.0
0.1
Number of local clusters increases their size decreases

Figure C.2.: Overview of the SynthGrid datasets used to evaluate LoGCC's scaling behavior. (a) depicts an ensemble with 4×4 local clusters, each consisting of 32×32 cells, resulting in a total ensemble size of $M = 128 \times 128$ cells and N members. (b) illustrates how increasing the scalar field size and the number of local clusters can be achieved by concatenating additional clusters. (c) demonstrates that subdividing a scalar field also increases the number of local clusters while decreasing their size. In both (b) and (c), reducing the proportion of unique clusters introduces global connections between local clusters.

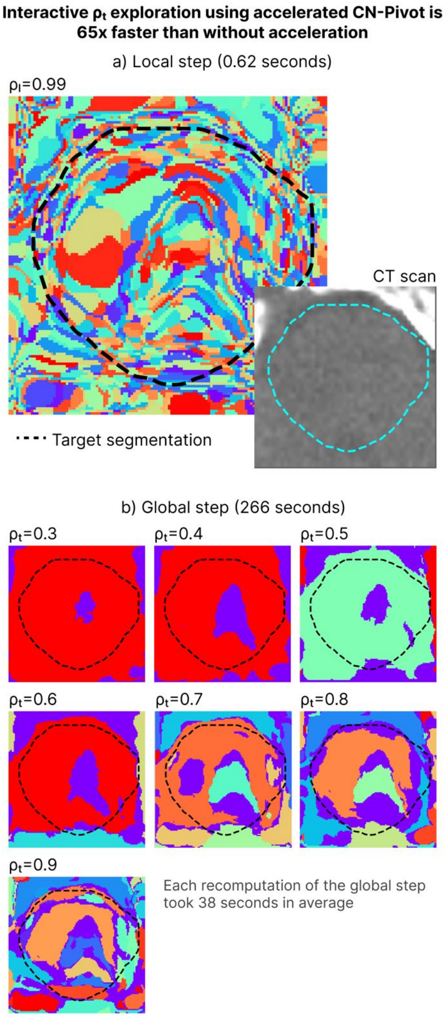

Interactive ρt exploration using accelerated CN-Pivot is 65x faster than without acceleration
a) Local step (0.62 seconds)
ρl=0.99
CT scan
Target segmentation
b) Global step (266 seconds)
ρt=0.3
ρt=0.4
ρt=0.5
ρt=0.6
ρt=0.7
ρt=0.8
ρt=0.9
Each recomputation of the global step took 38 seconds in average

Figure C.3.: Using the accelerated CN-Pivot method to explore the correlation structure of the HaN-Brainstem dataset using several correlation thresholds ($\rho_t \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$). The results of the local step (a) are re-used to compute several global clusterings in a fraction of the time (b). It takes the accelerated CN-Pivot method only 267 seconds to generate all the clusterings. In contrast, it takes the unaccelerated variant 17369 seconds, representing a 65x speedup. For context, we provide the CT scan slice for which the segmentation ensemble was computed and the target, ground truth, segmentation as a dashed line.

REFERENCES

[1] M. Cohen. *Practical Linear Algebra for Data Science: From Core Concepts to Applications Using Python*. O'Reilly, 2022.

ACKNOWLEDGEMENTS

At the end of September 2019, I had completed most of my Master's coursework and was ready to begin my graduation project. But first, I needed a project and a professor willing to supervise me. That was the purpose of my scheduled meeting with Klaus Hildebrandt and Anna Vilanova, the professors from the Geometric Data Processing and Data Visualization courses, respectively.

Due to a scheduling misunderstanding, I missed our first meeting—a moment of deep embarrassment. Would they still be willing to supervise me? Insecurity crept in, and I feared my future in visualization was over before it had even begun. Thankfully, I was overreacting. Klaus and Anna not only agreed to supervise my Master's thesis but, in late 2020, amid the COVID pandemic, they also helped me find my PhD position in the Human-Centric AI for Contouring in Head-and-Neck Cancer (HCAI) project.

Fast-forward a few years, and I am now completing my PhD—a feat that would not have been possible without the guidance and support of my supervisors, Klaus, Anna, and René van Egmond. Words fall short while expressing my gratitude, but let me try. Klaus, thank you for leading by example, demonstrating intellectual curiosity, and an unwavering work ethic. Anna, thank you for your sharp critiques, which always pushed me toward better work, and for infusing warmth and a personal touch into the research world. Your group at TU Eindhoven has been a second academic home. René, thank you for challenging my inclination to stay behind a screen and teaching me to empathize with users. Your human-centered approach significantly shaped our research, and I strive to embrace it every day.

Beyond my supervisory team, I am deeply grateful to my colleagues in the HCAI project. Huib, your relentless curiosity and openness to understand my work, even when outside your expertise, have been inspiring. Prerak, my academic brother—our conversations about research and life as PhDs kept me grounded and energized. Marius, a fantastic team leader. Thank you for standing up for us, even when it required difficult conversations.

I also acknowledge the institutions and individuals who played key roles in my research. Varian, a Siemens Healthineers Company, funded my PhD through the HollandPTC-Varian Consortium. At HollandPTC, I conducted user studies and benefited from the invaluable support of Jenneke, Charlotte, and Astrid. At LUMC, Eleftheria and Mischa provided insightful feedback. My heartfelt thanks also go to all the clinicians who dedicated time and energy to participate in our studies.

My experience was enriched by my colleagues at the CGV Group in Delft. I was fortunate to be part of such a diverse and inspiring environment, learning from experts in computer graphics, data visualization, computer vision, AI, and VR. Special thanks to Elmar for fostering a creative and hardworking culture and to Mathijs—what started as a Visulunch comment evolved into a full-blown research collaboration. Thanks also to Ruben and Mark, whose artistic contributions shaped this dissertation's cover. Ruben, your creativity and willingness to listen and help continually surprised and inspired me. Finally, thank you, Lauretta and Ruud, for ensuring the environment was set up so we could thrive. Not once did I have to think about an annoying process or a lack of work materials—everything was always taken care of. For that, I am very grateful.

Due to the interdisciplinary nature of my project, I often worked at the PILab at the Design Faculty, where René's office was. Thank you, Sylvia, for always making me feel part of the team. Yuguang, Stefano, and Gijs (Louwers), I loved our lunch and coffee conversations about, well, everything. Gijs (Huisman), I never thought I'd meet a bigger coffee geek than myself—thank you for our coffee discussions and for gifting me my first manual grinder, which has elevated my extractions ever since!

When Anna moved to TU Eindhoven, I thought it would distance me from data visualization. Instead, it introduced me to my second academic family, the NLVis group. Our post-COVID Rome conference was just the beginning of many great moments together. Thank you, Vidya, Astrid, Dennis, Alex, Julian, Sanne, Faizan, ChangLi, Linhao, Kirsten, and Marcos. Our "reading groups" were not only about research but also about getting to know each other beyond academia. Julian, since my Master's, you have been a constant creative force in my life—I look forward to seeing where your "computer whispering" skills take you.

I also want to acknowledge others who enhanced my academic journey. Eduard and Renata, thank you for being amazing hosts in Vienna. Eduard, I'll never forget my first daytime schnapps experience! Renata, I appreciated your generosity in showing me around and introducing me to your family. Adrien Bousseau, your feedback on my first visualization paper taught me the importance of storytelling in research. Kevin Hölhein and Rüdiger Westermann, thank you for sharing your code and datasets, which were instrumental in my work.

Some colleagues became friends, and some friends became family. Martin and Elisabeth, you were among the first people I met during my PhD—thank you for your advice and hospitality (and for all the pretzels!). Jan Jaap and Soyeon, our gin and tonic nights, great food, and Jan Jaap's studio-level photos always brought joy. Idil, you and Soyeon are my academic sisters—I can't wait for our next gossip session! Alex, not many of my friends have visited Colombia. But you have. I cherish memories like having our first coffee of the day while looking at the impressive mountains at Jardín.

Quality research thrives on a quality life outside academia. Over the years, I built a strong support network—my family in the Netherlands. Thank you to my Colombian gang (Fede, Ana, Juanda, Nadine, Lau, and Leo), Pierpa, Klara, Judy, Mauro, Ewoud and Giu, Sabine, and Janneke and Harry, who taught me how to make a proper espresso at their Café. Special thanks to Fede and Pierpa, the brothers I never had. Fede, our time as roommates at the 238 in my early PhD years was formative and so much fun. Pierpa, your unconventional thoughts on everything amuse and challenge me—let's keep debating them over beers around the world!

To conclude, I dedicate this dissertation to those whose importance is beyond words. To Ramona, my Mona. You are a spark of light. Your empathy, joy, and unwavering support have made life an easy and fulfilling journey. You and your family—Jojo, Henry, Gab, and Nobi—welcomed me as one of your own. For that, I am deeply grateful. I love you. To Josefa and Betty. You were as present in my upbringing as my parents, teaching me humility and the true meaning of hard work. I often long for the unencumbered days, when I would procrastinate homework by chatting over your delicious Colombian "onces". Finally, to my parents, Liliana (Lili) and Fernando (Fercho). This dissertation—indeed, my entire existence—would not be possible without you. Your love, sacrifices, and unwavering belief in me have shaped the person I am today. I owe you everything. Love you forever.

CURRICULUM VITæ

Nicolas Fernando Chaves de Plaza

1995	Born in Bogotá, Colombia.

EDUCATION

2007–2012	Secondary education Liceo Cervantes Norte, Bogotá, Colombia
2013–2017	Bachelor degree in Systems and Computing Engineering (Minor in Art History) Universidad de los Andes, Bogotá, Colombia
2013–2018	Bachelor degree in Management Universidad de los Andes, Bogotá, Colombia
2018–2020	MSc. Computer Science, Data Science & Technology Delft University of Technology, Delft, the Netherlands
2024	Research internship Vienna University of Technology, Vienna, Austria
2020–2024	PhD. Computer Graphics and Visualization Delft University of Technology, Delft, the Netherlands *Thesis:* Visual Analysis of Contour Ensembles in the Context of Radiotherapy *Promotors:* Dr. K. Hildebrandt, Dr. R. van Egmond and Prof. Dr. A. Vilanova

LIST OF PUBLICATIONS

In this dissertation:

5. **Chaves-de-Plaza, Nicolas F.**, R. G. Raidou, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "LoGCC: Local-to-Global Correlation Clustering for Scalar Field Ensembles". Submitted for publication. 2025

4. **Chaves-de-Plaza, Nicolas F.**, M. Molenaar, P. Mody, M. Staring, R. van Egmond, E. Eisemann, A. Vilanova, and K. Hildebrandt. "Depth for Multi-Modal Contour Ensembles". In: *Computer Graphics Forum* 43.3 (2024), e15083. doi: 10.1111/CGF.15083

3. **Chaves-de-Plaza, Nicolas F.**, P. Mody, M. Staring, R. van Egmond, A. Vilanova, and K. Hildebrandt. "Inclusion Depth for Contour Ensembles". In: *IEEE Transactions on Visualization and Computer Graphics* 30.9 (2024), pp. 6560–6571. doi: 10.1109/TVCG.2024.3350076

2. **Chaves-de-Plaza, Nicolas F.**, P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, A. Vilanova, and R. van Egmond. "Implementation of delineation error detection systems in time-critical radiotherapy: Do AI-supported optimization and human preferences meet?" In: *Cognition, Technology & Work* (2024). doi: 10.1007/s10111-024-00784-4

1. **Chaves-de-Plaza, Nicolas F.**, P. Mody, K. Hildebrandt, M. Staring, E. Astreinidou, M. de Ridder, H. de Ridder, and R. van Egmond. "Towards fast human-centred contouring workflows for adaptiv e external beam radiotherapy". In: *Proceedings of the Human Factors and Ergonomic s Society Europe*. 2022, pp. 111–31

Other publications:

4. P. Mody, M. Huiskes, **Chaves-de-Plaza, Nicolas F.**, A. Onderwater, R. Lamsma, K. Hildebrandt, N. Hoekstra, E. Astreinidou, M. Staring, and F. Dankers. "Large-scale dose evaluation of deep learning organ contours in head-and-neck radiotherapy by leveraging existing plans". In: *Physics and Imaging in Radiation Oncology* 30 (2024), p. 100572. doi: 10.1016/j.phro.2024.100572

3. M. Skrodzki, H. van Geffen, **Chaves-de-Plaza, Nicolas F.**, T. Höllt, E. Eisemann, and K. Hildebrandt. "Accelerating Hyperbolic t-SNE". in: *IEEE Transactions on Visualization and Computer Graphics* 30.7 (2024), pp. 4403–4415. doi: 10.1109/TVCG.2024.3364841

2. P. Mody, **Chaves-de-Plaza, Nicolas F.**, K. Hildebrandt, and M. Staring. “Improving Error Detection in Deep Learning Based Radiotherapy Autocontouring Using Bayesian Uncertainty”. In: *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*. Ed. by C. H. Sudre, C. F. Baumgartner, A. Dalca, C. Qin, R. Tanno, K. Van Leemput, and W. M. Wells III. Cham: Springer Nature Switzerland, 2022, pp. 70–79. doi: 10.1007/978-3-031-16749-2_7

1. P. Mody, **Chaves-de-Plaza, Nicolas F.**, K. Hildebrandt, R. d. R. van Egmond, Huib, and M. Staring. “Comparing Bayesian models for organ contouring in head and neck radiotherapy”. In: *Medical Imaging 2022: Image Processing*. Ed. by O. Colliot and I. Išgum. Vol. 12032. International Society for Optics and Photonics. SPIE, 2022, 120320F. doi: 10.1117/12.2611083