NATIONAL TECHNICAL
UNIVERSITY OF ATHENS
SCHOOL OF ELECTRICAL AND
COMPUTER ENGINEERING
DIVISION OF INFORMATION
TRANSMISSION SYSTEMS AND
MATERIAL TECHNOLOGY

Automated liver and liver tumor segmentation based on deep learning for 3D computed tomography of patients with colorectal liver metastasis

DIPLOMA THESIS

by

PAPANIKOLAS EMMANOUIL

Supervisor: George Matsopoulos
Professor, NTUA

Co-supervisor: Ourania Petropoulou
Permanent Laboratory Teaching Staff, NTUA

Athens, October 2025

This page is blank on purpose.

NATIONAL TECHNICAL
UNIVERSITY OF ATHENS
SCHOOL OF ELECTRICAL AND
COMPUTER ENGINEERING
DIVISION OF INFORMATION
TRANSMISSION SYSTEMS AND
MATERIALS TECHNOLOGY

Automated liver and liver tumor segmentation based on deep learning for 3D computed tomography of patients with colorectal liver metastasis

DIPLOMA THESIS

by

PAPANIKOLAS EMMANOUIL

Supervisor: George Matsopoulos
Professor, NTUA

Co-supervisor: Ourania Petropoulou
Permanent Laboratory Teaching Staff, NTUA

Approved by the three-member scientific committee on October 14, 2025.

. . . . .	. . . . .	. . . . .
George Matsopoulos	Athanasios Panagopoulos	Panayiotis Tsanakas
Professor, NTUA	Professor, NTUA	Professor, NTUA

Athens, October 2025

This page is blank on purpose.

(Signature)

.....

PAPANIKOLAS EMMANOUIL

Graduate of School of Electrical and Computer Engineering, National Technical University of Athens.

This page is blank on purpose.

Abstract

This project focuses on the development of a reliable and automated 3D segmentation pipeline for liver and colorectal liver metastases (CRLM) using computed tomography (CT) scans. The study combines a detailed literature review on CRLM and deep learning methods with an extensive experimental process implemented in the MONAI framework. Multiple architectures, strategies and parameters were explored, with SegResNet emerging as the most effective model for tumor segmentation. The proposed two-stage pipeline first segments the liver and then uses the liver mask to guide the tumor segmentation task.

Particular attention was given to preprocessing and data augmentation to address issues such as class imbalance, depth variation, and heterogeneous tumor appearances. Experiments were conducted on an optimized dataset, excluding patients with very few tumor slices, using Dice similarity coefficient (DSC), recall, precision, and surface distance metrics for evaluation. The final liver model achieved state-of-the-art performance with a Dice score of 0.968, while the tumor segmentation model reached 0.674 Dice, a competitive result given the difficulty of the task and the limited data and hardware resources.

Overall, the project demonstrates the potential of deep learning and 3D medical imaging for the accurate segmentation of CRLM, providing a solid foundation for future research using larger datasets and more specialized models.

Keywords: Computed tomography, Colorectal cancer, Colorectal liver metastasis, 3D image segmentation, Deep learning, Convolutional neural networks, U-Net, SegResNet

This page is blank on purpose

Περίληψη

Η παρούσα εργασία επικεντρώνεται στην ανάπτυξη ενός αξιόπιστου και αυτοματοποιημένου συστήματος τρισδιάστατης τμηματοποίησης (3D segmentation) ήπατος και ηπατικών όγκων από καρκίνο του παχέος εντέρου και του ορθού (CRLM), βασισμένο στη βαθιά μάθηση και με χρήση υπολογιστικών τομογραφιών (CT). Η μελέτη συνδυάζει μία εκτενή βιβλιογραφική ανασκόπηση για τον καρκίνο του παχέος εντέρου και τις ηπατικές μεταστάσεις, με μια πειραματική διαδικασία τμηματοποίησης με χρήση εργαλείων βαθιάς μάθησης. Δοκιμάστηκαν πολλαπλές αρχιτεκτονικές και στρατηγικές, με το δίκτυο SegResNet να αναδεικνύεται ως το πιο αποτελεσματικό για την τμηματοποίηση των όγκων. Η προτεινόμενη διαδικασία δύο σταδίων, ξεκινά με την τμηματοποίηση του ήπατος και στη συνέχεια χρησιμοποιεί την παραγόμενη μάσκα του ήπατος για να καθοδηγήσει την τμηματοποίηση των μεταστάσεων.

Ιδιαίτερη έμφαση δόθηκε στο στάδιο προεπεξεργασίας των δεδομένων και στις τεχνικές εμπλουτισμού δεδομένων, ώστε να αντιμετωπιστούν ζητήματα όπως η ανισορροπία των κλάσεων, η μεταβλητότητα του βάθους των εικόνων και η ετερογένεια των όγκων. Τα πειράματα πραγματοποιήθηκαν σε ένα βελτιστοποιημένο σύνολο δεδομένων, από το οποίο αποκλείστηκαν ασθενείς με υπερβολικά μικρό αριθμό όγκων και η αξιολόγηση πραγματοποιήθηκε με μετρικές όπως ο συντελεστής Dice (DSC), η ανάκληση (recall), η ακρίβεια (precision) και η απόσταση επιφανειών μεταξύ των μασκών. Το τελικό μοντέλο ήπατος πέτυχε επίδοση υψηλού επιπέδου με Dice 0.968, ενώ το μοντέλο όγκων έφτασε το 0.674, αποτέλεσμα ανταγωνιστικό με αντίστοιχη βιβλιογραφία, δεδομένης της πολυπλοκότητας του προβλήματος και των περιορισμένων πόρων.

Συνολικά, η εργασία αποδεικνύει τη δυναμική των μεθόδων βαθιάς μάθησης και της τρισδιάστατης ιατρικής απεικόνισης στην ακριβή τμηματοποίηση των ηπατικών μεταστάσεων από καρκίνο παχέος εντέρου και ορθού, προσφέροντας μια ισχυρή βάση για μελλοντική έρευνα με μεγαλύτερα σύνολα δεδομένων και πιο εξειδικευμένα μοντέλα και υποδομές.

Λέξεις-Κλειδιά: Αξονική τομογραφία, Καρκίνος παχέος εντέρου και ορθού, Ηπατικές μεταστάσεις καρκίνου παχέος εντέρου και ορθού, Τρισδιάστατη τμηματοποίηση εικόνων, Βαθιά μάθηση, Συνελικτικά νευρωνικά δίκτυα, U-Net, SegResNet

This page is blank on purpose

Acknowledgments

I would like to express my gratitude to my supervisor, professor Dr. George K. Matsopoulos, as well as to my co-supervisor, Ourania Petropoulou (Permanent Laboratory Teaching Staff, NTUA), for supervising the present thesis and for their participation in the evaluation process. I would also like to thank professors A. Panagopoulos and P. Tsanakas for their participation in the examination committee. In particular, I would like to thank Ph.D. candidate Mr. Theodoros P. Vagenas, who guided me during the preparation of this thesis and contributed decisively both to its implementation and to its writing. Finally, I express my sincere appreciation to my family, my girlfriend and my friends for their continuous support and encouragement throughout my studies.

Abbreviations and Acronyms

CNN	Convolutional Neural Network
CPU	Central Processing Unit
CRC	Colorectal Cancer
CRLM	Colorectal Liver Metastases
DL	Deep Learning
DSC	Dice Similarity Coefficient
GPU	Graphics Processing Unit
GT	Ground Truth
MI	Medical Imaging
ML	Machine Learning
MRI	Magnetic Resonance Imaging
PET	Positron Emission Tomography
ROI	Region of Interest
ViT	Vision Transformer

Contents

List of Figures

List of Tables

Εκτενής περίληψη στα ελληνικά

Εισαγωγή

Ο καρκίνος του παχέος εντέρου και του ορθού (Colorectal Cancer) είναι από τις πιο συχνές μορφές καρκίνου παγκοσμίως και παραμένει μια από τις πιο θανατηφόρες μορφές καρκίνου. Μελέτες αναφέρουν ότι περίπου το 10% των νέων διαγνώσεων καρκίνου κάθε χρόνο σχετίζεται με τον ΚΠΕΟ, ενώ μέχρι και το 50% των ασθενών έχει παρατηρηθεί ανάπτυξη μεταστάσεων στο ήπαρ (Colorectal Liver Metastases) σε περίπου 50% των ασθενών. Το ήπαρ αποτελεί το πιο συχνό όργανο μετάστασης του ΚΠΕΟ, λόγω της άμεσης αγγειακής σύνδεσής τους μέσω του πυλαίου φλεβικού συστήματος. Οι μεταστάσεις στο ήπαρ συνδέονται με πολύ χαμηλά ποσοστά επιβίωσης, ειδικά για διαγνώσεις σε προχωρημένο στάδιο. Η πενταετής επιβίωση των ασθενών που εντοπίζουν τον ΚΠΕΟ σε πρώιμο στάδιο φτάνει το 90%, ωστόσο για διαγνώσεις στο μεταστατικό στάδιο το ποσοστό πέφτει στο 14%.

Ο βασικός σκοπός της παρούσας εργασίας είναι η ανάπτυξη μιας ολοκληρωμένης μεθοδολογίας για την αυτόματη τρισδιάστατη τμηματοποίηση του ήπατος και των μεταστατικών όγκων από το παχύ έντερο, σε εικόνες αξονικής τομογραφίας (CT). Στόχος είναι η δημιουργία ενός αυτόματου συστήματος που θα μπορεί να υποστηρίξει τη διαδικασία διάγνωσης και εύρεσης της κατάλληλης θεραπευτικής στρατηγικής, καθώς η χειροκίνητη τμηματοποίηση όγκων είναι μια αρκετά χρονοβόρα διαδικασία. Η ακριβής τμηματοποίηση βοηθά στον καλύτερο σχεδιασμό του πλάνου αντιμετώπισης, στην εκτίμηση επαρκούς υπολειπόμενου υγιούς ήπατος σε περιπτώσεις αφαίρεσης της καρκινογόνας περιοχής, αλλά και στην παρακολούθηση της πορείας της νόσου.

Η ασθένεια

Ο ΚΠΕΟ ξεκινά συνήθως από καλοήθεις πολύποδες στο εσωτερικό τοίχωμα του εντέρου, οι οποίοι ενδέχεται στην πορεία να εξελιχθούν σε καρκίνο. Η νόσος χωρίζεται σε στάδια (0 έως IV), ανάλογα με το πόσο βαθιά στο τοίχωμα του εντέρου εντοπίζονται καρκινικά κύτταρα και αν υπάρχουν μεταστάσεις σε άλλα όργανα. Ο μεταστατικός ΚΠΕΟ στο ήπαρ ανήκει στο στάδιο IV (μεταστατικός καρκίνος). Τα πρώιμα στάδια έχουν καλύτερη πρόγνωση και θεραπευτική επιτυχία σε αντίθεση με τα μεταστατικά στάδια.

Η δημιουργία μεταστάσεων στο ήπαρ είναι ένα πολυσύνθετο βιολογικό φαινόμενο. Τα καρκινικά κύτταρα από το παχύ έντερο μεταφέρονται μέσω του αίματος στο ηπατικό παρέγχυμα, όπου αλληλεπιδρούν με τα κύτταρα του ήπατος και δημιουργούν το κατάλληλο «μικροπεριβάλλον» για να αναπτυχθούν νέοι όγκοι. Αυτή η άμεση σύνδεση των δυο οργάνων μέσω του καρδιαγγειακού συστήματος είναι και ο βασικός λόγος που το ήπαρ εμφανίζεται τόσο συχνά ως σημείο μετάστασης.

Η διάγνωση του μεταστατικού ΚΠΕΟ στο ήπαρ βασίζεται σε απεικονιστικές μεθόδους, με την αξονική τομογραφία να αποτελεί την πιο διαδεδομένη επιλογή. Η μαγνητική τομογραφία

(MRI) και η τομογραφία εκπομπής ποζιτρονίων (PET) παρέχουν επιπλέον πληροφορίες, αλλά συχνά δυσκολεύονται αν εντοπίσουν επαρκώς τους μιρούς μρταστατικούς όγκους. Οι θεραπευτικές επιλογές για τη συγκεκριμένη μετάσταση εξαρτώνται από την έκταση της. Σε μικρό αριθμό ασθενών (10–20%) είναι δυνατή η αφαίρεση των μεταστάσεων με χειρουργείο, που θεωρείται η πιο αποτελεσματική προσέγγιση. Ωστόσο σε περιπτώσεις αυξημένης προσβολης του ήπατος από τον καρκίνο δεν είναι η αφαίρεση μεγάλου έρους του οργάνου κρίνεται επικίνδυνη, οπότε εφαρμόζονται τοπικές τεχνικές αντιμετώπισης όπως θερμική ή ραδιοσυχνική κατάλυση, στοχευμένη ακτινοβολία, αλλά και άμεσες εγχύσεις χημειοθεραπείας στην ηπατική αρτηρία. Σε πιο σοβαρά περιστατικά, χρησιμοποιούνται συστηματικές θεραπείες όπως χημειοθεραπεία και ανοσοθεραπεία.

Οι μεταστάσεις του ήπατος παρουσιάζουν ορισμένα χαρακτηριστικά που βοηθούν τις απεικονιστικές μεθόδους να τις αναγνωρίσουν. Για παράδειγμα, συχνά εμφανίζουν υποαγγειακή εικόνα, δηλαδή φαίνονται λιγότερο έντονες από το φυσιολογικό ήπαρ μετά από έγχυση σκιαγραφικού στην αξονική. Παράλληλα το σχήμα τους παρουσιάζει έναν περιφερειακό δακτύλιο που οφείλεται σε νέκρωση των κυττάρων προς το κέντρο του όγκου. Ωστόσο πρόκειται για μια, κατα βάση, δύσκολα αναγνωρίσιμη μορφή καρκίνου, εξαιτίας της ποικιλομορφίας που εμφανίζει σε σχήμα και μέγεθος, καθώς και την έλλειψη αντίθεσης με το γειτονικό περιβάλλον του ήπατος στην τομογραφία.

Για να αντιμετωπιστούν οι παρπάνω δυσκολίες ανίχνευσης και ακριβούς τμηματοποίησης των μεταστατικών όγκων ΚΠΕΟ, έχουν χρησιμοποιηθεί σύγχρονες λύσεις με βάση την τεχνητή νοημοσύνη. Συγκεκριμένα, η βαθιά μάθηση (Deep Learning) έχει αποδειχθεί ιδιαίτερα αποτελεσματική στην επεξεργασία και ανάλυση ιατρικών εικόνων και μπορεί να βελτιώσει σημαντικά την ακρίβεια και την ταχύτητα της ανάλυσης εικόνων, μειώνοντας ταυτόχρονα τον φόρτο εργασίας και τα πιθανά λάθη των γιατρών.

Βαθιά Μάθηση

Η βαθιά μάθηση (Deep Learning) αποτελεί ένα από τα πιο σύγχρονα εργαλεία της τεχνητής νοημοσύνης, με αυξημένη εφαρμογή στην ιατρική απεικόνιση. Πρόκειται για μια προέκταση της μηχανικής μάθησης (Machine Learning), και αφορά τη δημιουργία μοντέλων που εκπαιδεύονται σε μεγάλα σύνολα δεδομένων με στόχο να ανακαλύψουν πρότυπα και σχέσεις μεταξύ των δεδομένων μεταξύ τους. Σε αντίθεση με τις κλασικές μεθόδους που απαιτούν ανθρώπινη παρέμβαση για την εξαγωγή χαρακτηριστικών, τα συστήματα αυτά μαθαίνουν αυτόματα τα πιο χρήσιμα χαρακτηριστικά απευθείας από τα δεδομένα.

Οι εφαρμογές της βαθιάς μάθησης χωρίζονται σε διάφορες κατηγορίες. Στον τομέα της ιατρικής απεικόνισης, οι πιο συχνές είναι:

- Ταξινόμηση (classification): αναγνώριση και κατηγοροιοποίηση εικόνων, π.χ. αν ένας όγκος είναι καλοήθης ή κακοήθης.
- Τμηματοποίηση (segmentation): ακριβής διαχωρισμός των ορίων ενός οργάνου ή ενός όγκου σε εικόνες CT/MRI.
- Εντοπισμός (detection/localization): εντοπισμός ύποπτων περιοχών με χρήση πλαισίων ή σημείων ενδιαφέροντος.
- Ανίχνευση ανωμαλιών (anomaly detection): ανάδειξη ασυνήθιστων προτύπων που διαφέρουν από το φυσιολογικό ιστό.

Ένα σημαντικό χαρακτηριστικό της ΒΜ είναι η συνοδεία των αρχικών δεδομένων από αντίστοιχες ετικέτες (labels). Στην εποπτευόμενη μάθηση (supervised learning), τα δεδομένα εκπαίδευσης διαθέτουν ετικέτες που περιλαμβάνουν τις κατηγορίες ή τα όρια που προσπαθεί να αναγνωρίσει το μοντέλο, ώστε να το καθοδηγήσουν στην διαδικασία παραγωγής των αντίστοιχων προβλέψεων. Στην μη εποπτευόμενη μάθηση (unsupervised learning), το σύστημα προσπαθεί να βρει κρυφά πρότυπα χωρίς τη βοήθεια ετικετών. Τέλος, η ημι-εποπτευόμενη μάθηση (semi-supervised learning), αποτελέι ένα συνδυασμό των παραπάνω εθόδων , όπου μόνο ένα μέρος των δεδομένων είναι επισημασμένο.

Τα νευρωνικά δίκτυα (Neural Networks) αποτελούν το βασικό εργαλείο της ΒΜ. Είναι μοντέλα εμπνευσμένα από τον ανθρώπινο εγκέφαλο, αποτελούμενα από στρώματα τεχνητών «νευρώνων» που λαμβάνουν σήματα, τα επεξεργάζονται και τα προωθούν στα επόμενα επίπεδα. Όταν τα δίκτυα αυτά αποκτούν πολλά κρυφά στρώματα, ονομάζονται βαθιά νευρωνικά δίκτυα (Deep Neural Networks – DNNs) και έχουν τη δυνατότητα να αφομοιώνουν πολύπλοκες αναπαραστάσεις και μοτίβα.

Ιδιαίτερη ρόλο στον τομέα επεξεργασίας εικόνων έχουν τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks). Περιλαμβάνουν ειδικά φίλτρα που εφαρμόζονται πάνω στην εικόνα ώστε να εντοπίζουν βασικά χαρακτηριστικά, όπως άκρα, σχήματα και στη συνέχεια πιο σύνθετες δομές. Τα ΣΝΔ έχουν αποδειχθεί εξαιρετικά αποτελεσματικά σε εφαρμογές ιατρικής απεικόνισης, καθώς μπορούν να αναγνωρίσουν και να απομονώσουν όγκους με μεγαλύτερη ακρίβεια από τις παραδοσιακές μεθόδους.

Η εκπαίδευση ενός μοντέλου βαθιάς μάθησης απαιτεί μεγάλο όγκο δεδομένων και υψηλή υπολογιστική ισχύ, κάτι που σήμερα επιτυγχάνεται με τη χρήση ειδικών καρτών γραφικών (GPUs). Ένα συχνό πρόβλημα είναι η υπερεκπαίδευση (overfitting), δηλαδή η υπερβολική προσαρμογή του μοντέλου στα συγκεκριμένα χαρακτηριστικά των δεδομένων εκπαίδευσης, που μειώνει την απόδοσή του σε νέα άγνωστα δεδομένα. Για να αποφευχθεί αυτό, χρησιμοποιούνται τεχνικές όπως η κανονικοποίηση (normalization), η ρύθμιση πολυπλοκότητας (regularization) και η επαύξηση δεδομένων (data augmentation).

Ένα σημαντικό στοιχείο των εργασιών ΒΜ είναι οι μετρικές αξιολόγησης (evaluation metrics) που ποσοτικοποιούν την απόδοση ενός μοντέλου. Στις εργασίες τμηματοποίησης, όπως στην παρούσα εργασία, οι πιο συχνές μετρικές είναι ο συντελεστής Dice (Dice Similarity Coefficient) και το Intersection over Union (IoU), που μετρούν πόσο καλά ταιριάζει η περιοχή που προβλέπει το μοντέλο με την πραγματική «μάσκα» του όγκου. Για προβλήματα εντοπισμού, συχνά χρησιμοποιούνται δείκτες όπως η ακρίβεια (precision), η ανάκληση (recall) και η μέση ακρίβεια (mean average precision) που υπολογίζουν τα ποσοστά ψευδώς θετικών και ψευδώς αρνητικών προβλέψεων. Οι μετρικές αυτές επιτρέπουν συγκρίσεις μεταξύ διαφορετικών μοντέλων και καθοδηγούν τη βελτίωση των στρατηγικών εκπαίδευσης.

Παράλληλα, ο βασικός μηχανισμός που καθοδηγεί τη εκμάθηση χαρακτηριστικών από το μοντέλο είναι οι συναρτήσεις κόστους (loss functions). Υπολογίζουν τη διαφορά ανάμεσα στην πρόβλεψη του μοντέλου και στην πραγματική τιμή, και καθοδηγούν το μοντέλο να διορθώσει τα βάρη του δικτύου κατάλληλα, αφού τιμωρούν τις λανθασμένες προβλέψεις. Στις εργασίες τμηματοποίησης όγκων συναντώνται συχνά η cross-entropy loss, η οποία εστιάζει στη σωστή ταξινόμηση κάθε pixel, και η Dice loss, που δίνει έμφαση στην ακριβή επικάλυψη των μασκών. Συχνά μάλιστα οι ερευνητές συνδυάζουν περισσότερες από μία συναρτήσεις κόστους ώστε να πετύχουν καλύτερη ισορροπία στην εκπαίδευση.

Μια ακόμα συχνή πρακτική που χρησιμοποιείται είναι η μεταφορά μάθησης (transfer learning), δηλαδή η αξιοποίηση μοντέλων που έχουν εκπαιδευτεί σε μεγάλα σύνολα δεδομένων (π.χ. εικόνες γενικής χρήσης) για γενικότερες εργασίες και η προσαρμογή τους σε πιο εξειδικευμένες εργασίες, όπως η ιατρική απεικόνιση συγκεκριμένων οργάνων ή ασθενειών. Αυτό βοηθά

ιδιαίτερα όταν τα διαθέσιμα δεδομένα είναι περιορισμένα, ωστέ το μοντέλο να μη ξεκινά την εκπαίδευση του χωρις καμία πρότερη γνώση.

Τμηματοποίηση στην Ιατρική Απεικόνιση

Η τμηματοποίηση (segmentation) αποτελεί μία από τις κυριότερες εφαρμογές της βαθιάς μάθησης στην ιατρική απεικόνιση, καθώς επιτρέπει τον ακριβή διαχωρισμό της περιοχής που καταλαμβάνουν οι επιλεγμένες ανατομικές δομές και παθολογικές περιοχές στις ιατρικές εικόνες. Στην περίπτωση των ηπατικών μεταστάσεων ΚΠΕΟ, η τμηματοποίηση είναι κρίσιμη για την αναγνώριση των ορίων και του μεγέθους του όγκου και μετέπειτα για την επιλογή της κατάλληλης θεραπευτικής στρατηγικής.

Η κύρια μέθοδος απεικόνισης που χρησιμοποιείται για τον εντοπισμό και την παρακολούθηση των μεταστάσεων είναι η Αξονική Τομογραφία (Computed Tomography). Η ΑΤ επιτρέπει τη λήψη λεπτομερών εικόνων του ήπατος με υψηλή ανάλυση, συνήθως σε πολλαπλές φάσεις (π.χ. αρτηριακή, φλεβική), κάτι που βοηθά στην καλύτερη διαφοροποίηση των μεταστατικών αλλοιώσεων από τον φυσιολογικό ιστό. Ωστόσο δε λείπουν οι προκλήσεις σχετικά με ακρίβεια της ανάλυσης, ειδικά για μικρές βλάβες ή για ασθενείς που έχουν υποβληθεί σε χημειοθεραπεία.

Στο πεδίο της τμηματοποίησης, οι δύο κύριες προσεγγίσεις είναι η σημασιολογική τμηματοποίηση (semantic segmentation), όπου κάθε στοιχείο κατατάσσεται σε μία κατηγορία (π.χ. υγιές ήπαρ, όγκος, φόντο), όπως ακριβώς στην παρούσα εργασία, αλλά και η τμηματοποίηση παραδειγμάτων (instance segmentation), που διακρίνει ξεχωριστά πολλαπλές βλάβες μέσα στην ίδια εικόνα.

Οι σύγχρονες μελέτες εμφανίζουν μια μετάβαση προς τρσδιάστατα μοντέλα τμηματοποίησης που μπορούν να αξιοποιούν όλο τον όγκο δεδομένων κάθε ΑΤ και όχι απλά μεμονωμένες τομές και να παράγουν αντίστοιχες τρισδιάστατες προβλέψεις, δίνοντας μια συνολική εικόνα για τα χαρακτηριστικά του όγκου.

Τα τρία βασικά στάδια των εργασιών τμηματοποίησης με εργαλεία ΒΜ είναι η προεπεξεργασία των δεδομένων, η εκπαίδευση του μοντέλου και η αξιολόγηση των πορβλέψεων. Στο πρώτο στάδιο εφαρμόζονται συχνά τεχνικές κλιμάκωσης της έντασης ώστε οι τιμές φωτεινότητας να βρίσκονται σε ομοιόμορφα εύρη, διευκολύνοντας το νευρωνικό δίκτυο να αναγνωρίσει τα κατάλληλα πρότυπα. Οι εντάσεις στην ΑΤ εκφράζονται σε μονάδες Hounsfield (Hounsfield Units ή HU), οι οποίες αντιστοιχούν στη διαφορετική απορρόφηση της ακτινοβολίας από κάθε ιστό (π.χ. το νερό έχει 0 HU, ο αέρας -1000 HU, τα οστά πάνω από +1000 HU). Η επιλογή του κατάλληλου εύρους της κλίμακας αυτής είναι ιδιαίτερα χρήσιμη για τη διάκριση μεταξύ υγιούς ηπατικού ιστού και μεταστατικών βλαβών.

Αρχιτεκτονικές και στρατηγικές τμηματοποίησης

Η πρότυπη αρχιτεκτονική νευρωνικών δικτύω που χρησιμοποιείται σε έργα ΒΜ για τμηματοποίηση ιατρικών εικόνων είναι το U-Net, που παρουσιάζεται στο Σχήμα 5.1. Το U-Net χαρακτηρίζεται από τη συμμετρική του δομή σε σχήμα U, που περιλαμβάνει ένα μονοπάτι συστολής (contracting path) ή κωδικοποιητή (encoder) για την εξαγωγή χαρακτηριστικών και ένα μονοπάτι επέκτασης (expanding path) ή αποκωδικοποιητή (decoder) για την ανακατασκευή της εικόνας. Τα λεγόμενα skip connections συνδέουν τα αντίστοιχα επίπεδα των δύο μονοπατιών, επιτρέποντας τη μεταφορά λεπτομερειών υψηλής ανάλυσης προς τον αποκωδικοποιητή για να συνδυαστούν με τις πληροφορίες που έχουν εξαχθεί στα βαθύτερα στρώματα. Αυτή η αρχιτεκτονική έχει αποδειχθεί ιδιαίτερα αποτελεσματική σε εργασίες τμηματοποίησης, ειδικά σε

περιπτώσεις όπου απαιτείται ακριβής εντοπισμός μικρών ή δύσκολα διακριτών δομών, όπως οι ηπατικές μεταστάσεις. Η αρχιτεκτονική αυτή μπορεί να ενισχυθεί με residual blocks που συνδεόυν διαφορετικά επίπεδα του κωδικοποιητή για να ενισχύσουν την αφομοίωση χαρακτηριστικών και να μειώσουν το πρόβλημα της εξαφάνισης των βαθμίδων (vanishing gradients).

Εκτός από το U-Net, εξετάστηκαν και άλλες αρχιτεκτονικές που περιλαμβάνουν διάφορες στρατηγικές βελτίωσης. Το SegResNet (Σχήμα 5.2 χρησιμοποιεί διαδοχικά residual blocks, που διευκολύνουν τη ροή των πληροφοριών μέσα στο δίκτυο και μειώνουν το πρόβλημα της εξαφάνισης των βαθμίδων (vanishing gradients). Το Attention U-Net εισάγει μηχανισμούς που επιτρέπουν στο δίκτυο να «εστιάζει» στις πιο σημαντικές περιοχές της εικόνας, φιλτράροντας θόρυβο ή άσχετες πληροφορίες. Τέλος, πιο πρόσφατες προσεγγίσεις, όπως τα Vision Transformers (ViTs), εφαρμόζουν αρχές από το πεδίο της επεξεργασίας φυσικής γλώσσας και βασίζονται σε μηχανισμούς προσοχής (self-attention). Επεξεργάζονται το σύνολο της εικόνας για να αφομοιώσουν συσχετίσεις μεταξύ απομακρυσμένων περιοχών της. Αν και απαιτούν συνήθως μεγαλύτερα σύνολα δεδομένων, έχουν δείξει ελπιδοφόρα αποτελέσματα και στον χώρο της ιατρικής απεικόνισης.

Στη συνέχεια έγινε σύγκριση διαφόρων μελετών που αφορούν εργασίες τμηματοποίησης τρισδιάστστων ιατρικών εικόνων. Οι δημοσιεύσεις που χρησιμοποιήθηκαν ανέπτυξαν παραλλαγές του U-Net, όπως το 3D U-Net ή το V-Net, καθώς και τεχνικές επαύξησης δεδομένων για να βελτιώσουν την απόδοση σε περιορισμένα datasets. Για να αντιμετωπιστεί το φαινόμενο μικρών και διάσπαρτων όγκων στα δεδομένα μερικές έρευνες προχώρησαν στην αφαίρεση ασθενών ή εικόνων που δε περιλάμβαναν όγκους, ενώ άλλες περιέλαβαν στις μετρικές την τμηματοποίηση του φόντου, γεγονός που οδήγησε σε υπερβολικά υψηλά αποτελέσματα. Παρατήρηθηκε πως οι περισσότερες μελέτες χρησιμοποίησαν εξελιγμένες κάρτες γραφικών αυξημένης χωρητικότητας. Τα δεδομένα προέκυψαν από διαδεδομένες βάσεις ιατρικών εικόνων, διαφορετικές από την παρούσα εργασία. Τα αποτελέσματα τμηματοποίησης συκωτιού κυμαίνονται κοντά στο 96-98%, ενώ για τους μεταστατικούε όγκους, οι πιο ακριβείς έρευνες αναφέρουν Dice 70-85%.

Μεθοδολογία

Στόχος της πειραμματικής διαδικασίας που ακολουθήσαμε ήταν η έυρεση των κατάλληλων παραμέτρων και αρχιτεκτονικών που θα οδηγούσαν στη δημιουργία ενός ακριβούς και αξιόπιστου μοντέλου τρισδιάστατης τμηματοποίησης ήατος και των μεταστατικών όγκων από ΚΠΕΟ. Για το σκοπό αυτό χρησιμοποιήθηκε εκτενώς η βιβλιοθήκη MONAI, που είναι ειδικά σχεδιασμένη για εφαρμογές βαθιάς μάθησης στην ιατρική απεικόνιση. Το MONAI παρείχε έτοιμες υλοποιήσεις αρχιτεκτονικών, μετασχηματισμών, συναρτήσεων κόστους και μετρικών αξιολόγησης, που επιτάχυναν σημαντικά την ανάπτυξη του μοντέλου και διευκόλυναν τις δοκιμές διαφόρων στρατηγικών.

Το dataset που χρησιμοποιήθηκε προέρχεται από τη βάση The Cancer Imaging Archive (TCIA) και αποτελεί το μεγαλύτερο διαθέσιμο σύνολο δεδομένων με προεγχειρητικές ΑΤ από ασθενείς με μεταστατικό ΚΠΕΟ στο ήπαρ. Αποτελείται από 197 ασθενείς με παθολογικά επιβεβαιωμένες μεταστάσεις. Κάθε περίπτωση ασθενή περιλαμβάνει τρισδιάστατες σαρώσεις στην πυλαία φάση, καθώς και αντίστοιχες μάσκες για τις περιοχές του ήπατος, των όγκων, των αγγείων και του μελλοντικού ηπατικού υπολείμματος. Αρχικά τα δεδομένα μετατράπηκαν από DICOM σε NIfTI μορφή, ώστε να είναι πιο συμβατά με τα εργαλεία μηχανικής μάθησης και στη συνέχεια χωρίστηκαν σε σύνολα εκπαίδευσης (70%), επικύρωσης (10%) και δοκιμής (20%), με σταθερή σύνθεση ασθενών ώστε να διασφαλιστεί η συγκρισιμότητα μεταξύ πειραμάτων.

Η προεπεξεργασία των δεδομένων αποτέλεσε σημαντικό στάδιο της εργασίας. Εφαρμόστηκαν βασικοί μετασχηματισμοί όπως η επαναδειγματοληψία (resampling) σε κοινές διαστάσεις voxel, και η κανονικοποίηση της έντασης των εικόνων με χρήση παραθύρου Hounsfield από -100 έως 200 HU, ώστε να καλύπτεται ο ηπατικός ιστός και οι μεταστατικές αλλοιώσεις, ενώ περιοχές χωρίς ενδιαφέρον όπως οστά και αέρας να αποκλείονται. Ειδικό βάρος δόθηκε στο πρόβλημα της ανισορροπίας δεδομένων, καθώς αρκετοί ασθενείς είχαν ελάχιστες εικόνες με όγκους, γεγονός που θα οδηγούσε το μοντέλο να εκπαιδευτεί κυρίως στην αναγνώριση του φόντου και να έχει χειρότερα αποτελέσματα. Για τον σκοπό αυτό χρησιμοποιήθηκε η τεχνική RandCropByPosNegLabeld του MONAI, που δημιουργεί έναν αριθμό τυχαίων δειγμάτων με υπο-περιοχές της αρχικής εικόνας, τα οποία περιέχουν όγκους ανάλογα με μια προκαθορισμένη πιθανότητα της τάξης του 80%. Έτσι, βελτιώνεται η ισορροπία των κλάσεων αι διασφαλίζεται ότι το μοντέλο θα συναντήσει επαρκή δείγματα όγκων ώστε να αφομοιώσει τα χαρακτηριστικά τους. Παράλληλα, εφαρμόστηκαν τεχνικές επαύξησης δεδομένων (data augmentation), όπως περιστροφές, μεγεθύνσεις, αναστροφές, προσθήκη θορύβου και αλλαγές στην αντίθεση, ώστε να αυξηθεί η ικανότητα γενίκευσης και η ανθεκτικότητα του μοντέλου σε παραλλαγές.

Για την εκπαίδευση δοκιμάστηκαν διάφορες αρχιτεκτονικές, με κεντρικό άξονα το τρισδιάστατο UNet. Η τρισδιάστατη έκδοση που χρησιμοποιήθηκε στην παρούσα εργασία, περιλαμβάνει τρισδιάστατες συνελίξεις που συμβάλλουν στην εκμετάλλευση χωρικών συσχετίσεων μεταξύ διαδοχικών τομών της ΑΤ. Επιπλέον εξετάστηκαν: το Attention U-Net, που ενσωματώνει μηχανισμούς προσοχής, το SegResNet, βασισμένο στο UNet, αλλά με διαδοχικά residual blocks για βαθύτερη εκμάθηση χωρίς απώλεια πληροφορίας και μοντέλα με transformers (όπως το UNETR και Swin-UNETR). Η επιλογή πολλαπλών αρχιτεκτονικών επέτρεψε τη συστηματική σύγκριση και την ανάδειξη πλεονεκτημάτων ή περιορισμών σε διαφορετικά μοντέλα.

Για την εκπαίδευση του μοντέλου ακολουθήθηκε μια συμβατική δομή κώδικα για εφαρμογές τρισδιάστατης τμηματοποίησης που προσαρμόστηκε στι ειδικές ανάγκες των δεδομένων μας. Σε κάθε εποχή υπολογίζονταν οι βασικές μετρικές κόστους και Dice για τα δεδομένα εκπαίδευσης και επικύρωσης, σε συνδυασμό με μετρικές αξιολόγησης (recall, precision κλπ.) ώστε να παρακολουθείται σε πραγματικό χρόνο η απόδοση του μοντέλου. Στο τέλος κάθε πειράμματος υπολογίζαμε τις αντίστοιχες μετρικές στα άγνωστα ως τότε δεδομένα δοκιμής. Παράλληλα, πειραματιστήκαμε και καταλήξαμε σε βασικές υπερπαραμέτρους του μοντέλου όπως η συνάρτηση ενεροποίησης 'softmax' αφού οι κλάσεις μας είναι αυτοαποκλειόμενες, η βελτιστοποίηση με τον αλγόριθμο Adam, η επιλογή του κοινότυπου ρυθμού μάθησης 1e-4 και οι τεχνικές regularization όπως weight decay (1e-5) και dropout (0.1), ώστε να αποφευχθεί η υπερεκπαίδευση. Η αξιολόγηση βασίστηκε κυρίως στο Dice Similarity Coefficient (DSC), με τον υπολογισμό να επικεντρώνεται στην κλάση του όγκου, καθώς η εύκολη αναγνώριση του φόντου θα μπορούσε να «φουσκώσει» τεχνητά τα αποτελέσματά μας.

Παράλληλα, εξετάστηκε σύντομα και η στρατηγική της μεταφοράς μάθησης (transfer learning). Χρησιμοποιήθηκε ένα προεκπαιδευμένο μοντέλο U-Net από το MONAI Zoo για τμηματοποίηση σπλήνας που προσαρμόστηκε στα δεδομένα μας. Τα προεκπαιδευμένα βάρη είναι ικανά να επιταχύνουν τη σύγκλιση και να βελτιώσουν την απόδοση σε μικρότερα ή πιο δύσκολα datasets, όπως το παρόν. Παρόμοια πειράματα έγιναν και με τη SegResNet αρχιτεκτονική, που προεκπαιδεύτηκε σε τμηματοποίηση ήπατος από τα ίδια τα δεδομένα μας και στη συνέχεια βελτιστοποιήθηκε για τον εντοπισμό των όγκων.

Για τους ελέγχους στο άγνωστο σύνολο αξιολόγησης, τα βάρη των μοντέλων φορτώθηκαν και χρησιμοποιήθηκαν για την παραγωγή προβλέψεων. Οι τελικές τιμές DSC στο σύνολο αξιολόγησης, σε συνδυασμό με οπτικοποιήσεις των προβλέψεων χρησιμοποιήθηκαν για να επιλεγεί η καλύτερη αρχιτεκτονική για τις δυο εργασίες τμηματοποίησης. Τα καλύτερα μοντέλα στην τμηματοποίηση συκωτιού και όγκων συνδυάστηκαν στην πορεία σε ένα αυτόματο σύστημα

τμηματοποίησης, όπου οι προβλέψεις του συκωτιού χρησιμοποιήθηκαν για να καθοδηγήσουν το μοντέλο τμηματοποίησης των όγκων. Με αυτόν τον τρόπο, το μοντέλο αξιοποιεί χωρική πληροφορία και δομικά συμφραζόμενα, περιορίζοντας τα σφάλματα εντοπισμού και βελτιώνοντας την ακρίβεια στις περιοχές ενδιαφέροντος.

Αποτελέσματα και ανάλυση

Τμηματοποίηση ήπατος

Για την εργασία τμηματοποίησης ήπατος δοκιμάστηκαν οι αρχιτεκτονικές UNETR, SegResNet και ResUNet. Η διαδικασία προεπεξεργασίας ήταν απλή, χωρίς χρήση επαυξήσεων και το τελικό μέγεθος των εικόνων εισόδου ήταν 128-128-48. Το SegResNet είχε τη υψηλότερη απόδοση στο σύνολο αξιολόγησης με τελικό DSC 0.968, Recall 0.964, Precision 0.96 και καλύτερο DSC συνόλου επικύρωσης 0.97. Τα άλλα δυο μοντέλα είχαν ικανοποιητική απόδοση, αλλά λίγο χαμηλότερα τελικά αποτελέσματα, με DSC 0.963 για το ResUNet και 0.956 για το UNETR.

Τα αποτελέσματα του καλύτερου μοντέλου που παρουσιάζονται στον Πίνακα 7.3, είναι συγκρίσιμα με state-of-the-art μοντέλα τμηματοποίησης ήπατος. Παρατηρήθηκε υψηλή απόδοση όλων των αρχιτεκτονικών με παρόμοια τελικά αποτελέσματα, κάτι που αποδίδεται στην ομοιομορφία της περιοχής ενδιαφέροντος (μεγάλο, σαφώς οριοθετημένο όργανο), και τη σταθερότητα εμφάνισης του σε όλες τις εικόνες. Τα μοντέλα συγκλίνουν γρήγορα κατά την εκπαίδευση, με υψηλές τιμές Dice μετά από λίγες εποχές. Οι τεχνικές προ-επεξεργασίας φέρεται να σταθεροποίησαν την εκπαίδευση, αφού δε παρατηρήθηκε υπερπροσαρμογή.

Τμηματοποίηση μεταστατικών όγκων

Από τα αρχικά πειράματα καταλήξαμε στην εξαίρεση του φόντου από τον υπολογισμό της μετρικής Dice, και στη χρήση του βελτιστοποιημένου συνόλου δεδομένων με ασθενείς με περισσότερες από 10 τομές με παρουσία όγκου, αφού η σύγκλιση του μοντέλου ήταν πιο σταθερή και γρήγορη. Η σύγκριση των αρχιτεκτονικών ανέδειξε ως καλύτερο ξανά το μοντέλο SegResNet με τελικό DSC στο σύνολο αξιολόγησης 0.652, Recall 0.75 και Precision 0.614, όπως παρουσιάζονται στον Πίνακα 7.6. Τα μοντέλα AttentionUNet και ResUNet είχαν επίσης ικανοποιητική απόδοση με τελικό DSC 0.641 και 0.604, ενώ το μοντέλο μετασχηματιστών, SwinUNETR, ήταν αρκετά χειρότερο με τελικό αποτέλεσμα 0.525.

Το πείραμα μεταφοράς μάθησης από ένα μοντέλο SegResNet εκπαιδευμένο στην τμηματοποίηση ήπατος στο σύνολό μας έδειξε καλά αποτελέσματα, ωστόσο ενέχει κινδύνους υπερπροσαρμογής και δε προτιμήθηκε. Αντίστοιχα, η μεταφορά μάθησης από ένα UNet εκαπιδευμένο στην τμηματοποίηση συκωτιού είχε φτωχό αποτέλεσμα με τελικό σκορ 0.531, δείχνοντας αδυναμία γενίκευσης των χαρακτηριστικών της σπλήνας σε ετερογενείς μεταστατικούς όγκους του συκωτιού.

Τελικό αυτόματο σύστημα τμηματοποίησης

Με βάση τα παραπάνω αποτελέσματα, το SegResNet επιλέχθηκε ως αρχιτεκτονική του τελικού συστήματος, τόσο για την τμηματοποίηση του ήπατος, όσο και για την τμηματοποίηση των μεταστατικών όγκων. Η δομή του αυτοματοποιημένου συστήματος περιγράφεται στο Σχήμα 7.17. Οι προβλέψεις που παρήχθησαν για την περιοχή του συκωτιού χρησιμοποιήθηκαν τόσο ως δεύτερο κανάλι εισόδου για την ενίσχυση των χωρικών πληροφοριών του συστήματος

των όγκων, όσο και κατά την διάρκεια προεπεξεργασίας για την μεταφορά της περικοπή της αρχικής εικόνας γύρω αό την περιοχή του συκωτιού. Τα τελικά αποτελέσματα που παρουσιάζονται στον Πίνακα 7.12 ήταν αρκετά υποσχόμενα, αφού το τελικό DSC του συνόλου αξιολόγησης έφτασε το 0.674. Παράλληλα, τα αποτελέσματα των μετρικών αξιολόγησης ήταν 0.76 recall, 0.647 precision, 0.66 συνολικό μέσο DSC από όλους τους ασθενείς και 8.61 μέση απόσταση επιφανειών. Παρατηρήθηκε πως 20/22 ασθενείς είχαν τελικό DSC μεγαλύτερο του 40%, με την ύπαρξη δυο ασθενών με ακραίες τιμές, χωρίς τους οποίους το τελικό DSC ανέβαινε στο 0.728.

Οι οπτικοποίηση και σύγκριση των προβλέψεων του μοντέλου με τις ετικέτες που δόθηκαν, ανέδειξαν κάποια σημαντικά στοιχεία. Το μοντέλο φάνηκε να αποδίδει εξαιρετικά σε μεταστατικούς όγκους με ξεκάθαρο σχήμα, ικανοποιητικό μέγεθος και επαρκή αντίθεση από το περιβάλλον του συκωτιού, όπως στα Σχήματα 7.23 και 7.24. Αντίθετα, το μοντέλο φάνηκε να δυσκολέυεται να αναγνωρίσει ή να προβλέψει επαρκώς το σχήμα και το μέγεθος σε εικόνες με μειωμένη ποιότητα και ελλιπή αντίθεση, αλλά και σε πολύ μικρούς όγκους, όπως στα Σχήματα 7.20 και 7.21. Τέλος, υπήρξαν περιπτώσεις όπου οι ετικέτες ήταν ασταθείς ή και τελείως άστοχες, με αποτέλεσμα οι μετρικές να τιμωρούν το μοντέλο, ενώ έκανε ικανοποιητική πρόβλεψη.

Σύνοψη

Συνολικά, ο ΚΠΕΟ και οι ηπατικές μεταστάσεις του αποτελούν ένα σοβαρό ιατρικό πρόβλημα με αυξανόμενη σημασία. Η ανάπτυξη μεθόδων αυτόματης τμηματοποίησης με τη βοήθεια της τεχνητής νοημοσύνης ανοίγει τον δρόμο για πιο ακριβή και πιο αξιόπιστη διάγνωση, που μπορεί να βελτιώσει την κλινική πράξη και την ποιότητα ζωής των ασθενών.

Η μεθοδολογία μας στηρίχθηκε σε ένα προσεκτικά δομημένη στρατηγική, που συνδύασε εκτεταμένη προεπεξεργασία και ενισχυτικές τεχνικές, καθώς και την παράλληλη αξιολόγηση διαφορετικών αρχιτεκτονικών με σύγχρονες προσεγγίσεις. Σημαντικό κομμάτι αποτέλεσε η προσπάθεια διασφάλισης της αντικειμενικότητας των αποτελεσμάτων, με αποφυγή στοχευμένης εξαίρεσης ασθενών, ή λανθάνουσας βελτίωσης των αποτελεσμάτων με χρήση του φόντου. Το τελικό αποτέλεσμα τμηματοποίησης συκωτιού είναι συγκρίσιμο με καταξιωμένες έρευνες στην περιοχή, ενώ το αποτέλεσμα του αυτοματοποιημένου συστήματος τμηματοποίησης μεταστατικών όγκων, είναι πολλά υποσχόμενο, ειδικά δεδομένης της δυσκολίας τμηματοποίησης του συγκεκιρμένου τύπου καρκίνου, αλλά και των ασυνεπειών που παρατηρήθηκαν στα δεδομένα. Παράλληλα, δόθηκε έμφαση στη μείωση των ψευδώς αρνητικών (FN) προβλέψεων, που είναι κρίσιμες σε ιατρικές εφαρμογές. Ο συνδυασμός των χωρικών πληροφοριών πρόβλεψης συκωτιού και των τοπικών χαρακτηριστικών σχήματος, αντίθεσης και έντασης από την αρχική εικόνα στο αυτοματοποιημένο σύστημα κρίθηκε επιτυχημένος. Η αυξημένη απόδοση για ευκρινείς όγκους, επαρκούς μεγέθους και αντίθεσης αναδεικνύει τις δυνατότητες του μοντέλου.

Κατά τη διάρκεια της μελέτης αναδείχθηκαν ορισμένες προκλήσεις που επηρέασαν την εκπαίδευση και την απόδοση των μοντέλων. Συγκεκριμένα, ο απαιτούμενος χρόνος εκπαίδευσης ήταν αυξημένος, καθιστώντας αναγκαία τη χρήση ισχυρών GPU με επαρκείς υπολογιστικούς πόρους. Επιπλέον, παρατηρήθηκε ανισορροπία στο σύνολο δεδομένων, τόσο ως προς το βάθος των τομογραφιών όσο και ως προς τη συχνότητα εμφάνισης των όγκων. Οι ανακρίβειες στις ετικέτες, οι διαφορές στην ένταση και ποιότητα των εικόνων, καθώς και η μορφολογική ανομοιογένεια των μεταστατικών όγκων, αποτέλεσαν κρίσιμους παράγοντες που δυσχέραναν την εκπαίδευση και αξιολόγηση των μοντέλων.

Για να αντιμετωπιστούν οι παραπάνω περιορισμοί, προτείνονται πιο πλούσια και αντιπροσωπευτικά σύνολα δεδομένων, με καλύτερη επισημείωση από ειδικούς για πιο άμεση σύνδεση με κλινικές εφαρμογές και πιο ακρινή αποτελέσματα. Παράλληλα, η χρήση προχωρημένων αρ-

χιτεκτονικών που εξελίσσουν τις υπάρχουσες με στοχευμένες παραλλαγές και αλλά και σύγρονων στρατηγικών μετασχηματισμού και προσοχής, μπορούν να βελτιώσουν την ακρίβεια της τμηματοποίησης. Αντίστοιχα, εξειδικευμένες στατηγικές εκπαίδευσης, στοχευμένες στις ανάγκες του κάθε έργου τμηματοποίησης, μπορούν να ενισχύσουν την ακρίβεια και τη γενίκευση. Τέλος, η αναβάθμιση της υπολογιστικής υποδομής (σύγχρονες GPU, περιβάλλοντα νέφους), σε συνδυασμό με αυτοματοποιημένα και ευέλικτες συνδυαστικά συστήματα τμηματοποίησης, θα επιτρέψουν πιο αποδοτική και κλιμακώσιμη εκπαίδευση.

Part I

Theoretical Background and Bibliographic Review

Chapter 1

Introduction

1.1 Background

Colorectal cancer (CRC) is one of the most common types of cancer worldwide and a quite fatal type of cancer. Around 30-50% of patients with CRC eventually develop colorectal liver metastases (CRLM), as the liver is the most common site of distant spread due to its vascular connection with the colon and rectum. CRLM lesions have poor prognosis statistics and are often unresectable at diagnosis. The main treatment strategies include surgical resection, if possible, chemotherapy or alternative radiation therapies, which are determined by the size and distribution of metastatic lesions.

Clinical decisions regarding the diagnosis and treatment options for CRLM patients rely heavily on accurate CRLM imaging and analysis, usually through computed tomography (CT). Tumor segmentation is the process of separating the malignant metastatic region from the surrounding liver tissue, which can provide valuable insights into surgical planning, future liver remnant estimation and treatment planning.

Manual segmentation from expert radiologists is a time-consuming and costly process. That is why automatic segmentation techniques that leverage deep learning methods for image processing have been developed and applied in medical imaging in recent years. However, CRLM is a challenging type of cancer with small size, heterogeneity, and irregular shapes. As a result, there is a lack of complete and well-annotated datasets focused on CRLM lesions, and therefore a limited availability of deep learning models specifically designed for this task.

1.2 Research objectives and report structure

The primary objective of this project is the creation of a complete 3D segmentation pipeline for CRLM segmentation from CT scans. A two-stage automated pipeline is used for this challenging segmentation task. The liver segmentation precedes in order to guide the tumor segmentation towards the anatomically relevant region of the liver. Automating this process is quite important for future clinical application, where rapid and reliable results are crucial for timely diagnosis and treatment planning.

For this purpose, we divide the objective in two separate segmentation tasks, liver and tumor segmentation. The best models for each one are then combined in an automated segmentation pipeline that produces the final CRLM predictions. A variety of preprocessing techniques and augmentations is applied to the data to prepare it for the training

phase. Several prominent network architectures and hyperparameter configurations for 3D medical imaging segmentation are tested on each task. The results are then visualized through plots and tables, which help compare each processing and training strategy and architecture and finally come down with the best performing model for liver and tumor segmentation respectively.

The secondary objective is the development of a comprehensive literature review that provides the theoretical basis for the experimental part. This review begins with an overview of the disease, expanding on the development mechanisms and stages of colorectal cancer, along with the metastatic mechanism to the liver and the prevailing diagnostic and treatment options. The review continues with the presentation of key deep learning concepts, such as convolutional neural networks, evaluation metrics and training technicalities. Next, there is an introduction to the key concepts of segmentation tasks in medical imaging, including modalities, annotation standards and main challenges. The review ends with a chapter focusing on the most common network architectures used for segmentation tasks in medical imaging and a comparative analysis of strategies and networks used in similar research papers.

Our report continues with the presentation of the methodology and the experimental results and concludes with the critical assessment of the proposed approach and suggestions for further research in this field.

1.3 Scope and limitations

This project specifically focuses on the implementation of a 3D segmentation model for CRLM patients using a dataset consisting only of CT scans. Other modalities such as MRI and PET scans are not covered. Both the literature review and the experimental phase are limited to CRLM and do not cover other types of liver cancer, such as hepatocellular carcinoma, or CRC metastasis in different organs.

This is strictly a segmentation task and does not involve detection or classification processes. Many of the programming components for the creation and training of the segmentation model are imported from libraries like MONAI (Medical Open Network for Artificial Intelligence) and PyTorch, rather than being implemented entirely from scratch. This includes network architectures, preprocessing transforms, plotting modules and some metrics.

The patient data used is restricted to the specific public dataset provided by the instructors for this task, with no other open-source datasets incorporated. The local GPU available had only 4 GB of memory, which was not ideal for heavy 3D data and resulted in longer training and testing times. Later, remote access to a machine with a 16 GB GPU was granted, enabling faster and more specialized experimentation. However, there were substantial time and computational constraints that prevented the exploration of some available models, especially heavier ones, as well as the implementation of specialized deep learning techniques and architectures that were covered by relevant research.

Chapter 2

Medical Background on CRLM

2.1 Overview of Colorectal Cancer

2.1.1 General definition

A malignant tumor that develops from the inner lining of the colon or rectum is called colorectal cancer (CRC). It usually begins as a benign polyp and progresses through a number of histological, morphological, and genetic changes over several years of being asymptomatic, before finally becoming an invasive cancer. Anomalies known as colorectal polyps develop on the colon's or rectum's inner wall. Although the majority of polyps are benign and not cancerous, many of them are precancerous (adenomas), which can develop into cancer if they are left untreated for more than 5–7 years. The stage at diagnosis has a significant impact on the prognosis, as early detection through screening is essential to decrease incidence and mortality probabilities [12].

CRC can often spread to other organs through the lymphatic system or bloodstream, in a process known as metastasis. The liver is the most common organ for metastasis of CRC, due to its connection to the colon and rectum through the portal vein system. Despite advances in treatment methods, up to 50% of CRC patients develop CRLM, which is deemed the primary cause of death from CRC [13].

2.1.2 Current global epidemiology

Almost 10% of all annually diagnosed cancers and cancer-related deaths worldwide are related to CRC. It is the third most common cancer in men and second in women with an incidence and mortality rate of 25% lower than in men [14]. Due to advanced detection tools as well as dietary and lifestyle factors, developed countries have the highest incidence and mortality rates. On the other hand, urbanization and western globalization are causing rising trends in developing nations. Notably, there has been a worrying rise in the incidence of CRC in people under 50, particularly for left-sided and rectal cancers. Over 148,000 new cases and almost 50,000 deaths from CRC were predicted for a single year in the United States, with the 5-year survival rate being 90% for early detection, but drastically lower for detection at a regional or distant stage [15].

2.1.3 Risk factors

There are two main categories of risk factors for CRC: modifiable and non-modifiable. Age is one of the major non-modifiable risk factors, as the chance of developing CRC increases significantly over 50 years of age. Another contributing factor is sex, since the disease is more common in men than in women. A family history of CRC, especially involving first-degree relatives, is known to double an individual's lifetime risk of developing the condition [16]. Hereditary syndromes, such as Lynch syndrome and familial adenomatous polyposis are responsible for 5–7% of all CRC instances [13]. Type 2 diabetes and specific ethnic backgrounds are additional non-modifiable factors that have been linked to an increased risk of colorectal cancer.

The main modifiable risk factors include lifestyle and environmental parameters. Smoking, excessive alcohol consumption, obesity, and increased red and processed meat consumption are the main environmental factors related to this type of cancer [12]. Recent studies [14] suggest the potential role of gut microbiota, with bacterial species implicated in promoting colorectal carcinogenesis. On the other hand, specific diets, such as those high in calcium, green leafy vegetables, and fiber, have been found to lower the risk of CRC. Even with the acknowledgment and consideration of the above precautionary factors, screening remains essential for early detection and dramatically lowers the development and mortality rates of the disease.

2.2 Pathophysiology of CRC

2.2.1 Origin and early development of CRC

CRC typically develops from focal changes within benign precancerous polyps. These polyps are localized growths or aggregations of abnormal cells found within the inner lining of the colon or rectum [12]. Most polyps are benign (noncancerous), but certain types can change into cancer over the course of several years, making them quite common, especially in older ages.

Adenomas and sessile serrated polyps (SSPs) are the two main types with potential of becoming malignant. Adenomatous polyps are considered a precancerous condition because they can transform into cancer. The most common type is tubular adenomas. Villous adenomas are rarer, but more dangerous in terms of malignant transformation. The third type is tubulovillous adenomas, which exhibit mixed characteristics from the other two types. All adenoma types are characterized by dysplasia, meaning they exhibit abnormal growth, while still being benign.

On the other hand, SSPs are a type of colon polyp with a serrated or saw-toothed appearance under a microscope [12]. They are responsible for only 25% of colon cancer cases; however, these are generally more aggressive types of cancer. SSPs are classified in four categories based on their shape:

- **Hyperplastic polyps and inflammatory polyps** are more common, but in general they are not precancerous.
- **Sessile serrated lesions (SLLs)** are the most common precancerous polyps and have a flat shape, making them similar to hyperplastic polyps.

- **Traditional serrated adenomas** are the rarest type of serrated polyps, found in less than 1% of the population. They resemble traditional adenomas and are also precancerous.
- **Unclassified polyps** are serrated polyps that may appear sessile and serrated but also have signs of dysplasia or features resembling adenomas.

Several characteristics of polyps can increase the chances of malignant transformation, including size larger than 1 cm, number greater than three, and signs of dysplasia.

Most colorectal cancers are adenocarcinomas, which originate in cells that make mucus to lubricate the inside of the colon and rectum. Other, less common colorectal tumors include:

- Carcinoid tumors (from hormone-producing cells)
- Gastrointestinal stromal tumors (GISTs)
- Lymphomas (cancers of immune cells)
- Sarcomas (connective tissue cancers)

2.2.2 Stages

Over time, cancer that starts as a polyp may spread to the colon or rectum's wall, which is made up of several layers. The mucosa, the innermost layer, is where colorectal cancer begins, and it can spread through some or all of the other layers. Cancer cells in the wall have the potential to develop into lymphatic or blood vessels, which are microscopic channels that remove fluid and waste. They can then proceed to distant areas of the body or to neighboring lymph nodes. The stage (extent of spread) of a colorectal cancer depends on how deeply it grows into the wall and if it has spread outside the colon or rectum.

Colorectal cancer is diagnosed in five stages from 0 to IV with an increasing degree of spreading. Cancer can spread to other areas in the body through nearby tissue, blood, and the lymph system [17]. Several diagnostic tests are used to determine the stage of colorectal cancer, including biopsy, blood work, biomarker testing, chest X-ray, CT scan, MRI and PET scan. Based on the thorough description and presentation of the stages by the American Cancer Society, here are the characteristics of each one.

Stage 0

In stage 0 colorectal cancer, abnormal cells are found in the innermost layer of the colon or rectum, called the mucosa. There are two types of surgery suggested for this stage, polypectomy, where cancerous polyps are removed with a wire loop during colonoscopy, or local excision, which removes polyps from the colon lining along with a small amount of healthy tissue.

Stage 1

Stage I colorectal cancer has surpassed the mucosa and has spread into the muscular layer of the colon or rectum. Cancerous cells have been found in the mucosa, the second layer

(submucosa) and probably the third layer (muscularis propria), however, the disease has not spread to any lymph nodes or nearby tissue. The surgical options suggested for this stage are the same as for stage 0. Additional surgery for further tissue removal might be required for high grade cancerous polyps. If the cancer was not in a polyp, a partial colectomy is required to remove the cancerous portion of the colon and any nearby lymph nodes.

Stage 2

In stage II, the cancerous cells have spread into the outer layers of the colon or rectum but have not yet reached the lymph nodes or any other organs. Stage II colon cancer is divided into three categories.

- Stage IIA cancer has spread up until the muscularis propria layer of the colon.
- Stage IIB cancer has spread through to the outermost layer of the colon wall, called the serosa.
- Stage IIC colon cancer has spread through the colon wall and into nearby tissue.

Partial colectomy, a surgery that removes the section of colon where the cancer is located, as well as nearby lymph nodes, is the only treatment usually required at this stage. In some cases, adjuvant chemotherapy, hence chemo treatment given after the surgery, is recommended to help destroy any remaining cancer cells and reduce recurrence probabilities.

Similarly to colon cancer, stage II rectal cancer is also divided into three categories.

- Stage IIA rectal cancer has spread to the outermost layer (serosa) of the rectal wall, through the muscle layer of the rectum.
- In stage IIB the cancerous cells have spread through the serosa of the rectum to the tissue that wraps around the organs, called the visceral peritoneum, in the abdomen.
- Lastly, stage IIC rectal cancer has spread through the serosa to nearby organs.

According to the specific requirements of each case, a combination of chemotherapy, surgery and radiation is used to deal with stage II rectal cancer. Chemotherapy (usually 5-FU or capecitabine) and radiation are usually the primary treatment options, with a goal of reducing the size of the tumor ahead of surgical treatment. The final phase of treatment is additional chemo rounds after surgery.

Stage 3

Stage III colon cancer is characterized by tumor growth beyond the inner lining of the colon and its spread to nearby lymph nodes, but without further metastasis. In Stage IIIA, cancer has penetrated the innermost layers of the colon wall (the mucosa and submucosa) and has spread to nearby lymph nodes or surrounding tissue. Stage IIIB regards deeper invasion into the muscle layer, the outermost layer (serosa), or even into the lining of the abdominal cavity (visceral peritoneum), with involvement of up to six lymph nodes. In Stage IIIC, cancer has penetrated the serosa or nearby organs and has

spread to four or more lymph nodes. This stage signals advanced local disease and a significantly increased risk of distant metastasis, particularly to the liver.

The designated treatment to remove the infected section of the colon, along with nearby lymph nodes, at this stage of colon cancer is partial colectomy. For better results, chemotherapy regimens with a combination of drugs are used to accelerate the destruction of cancer cells. However, in cases where the tumor cannot be removed completely by surgery, chemotherapy along with radiation therapy are used to shrink the cancer so it can be removed later with surgery. In other cases where the cancer was attached to a nearby organ or the tissue that was removed had margins positive in cancer, radiation therapy may also be used after surgery to ensure complete healing of the region.

Stage 4

In stage IV colorectal cancer, the cancer has been carried through the lymph and blood systems to distant parts of the body. This type of cancer is called metastatic, since the cancer cells have formed a new tumor outside their place of origin. The new, metastatic tumor is the same type of cancer as the primary tumor, even though it is in a different part of the body. The most likely organs to develop metastasis from colorectal cancer are the lungs and liver. Stage IV colon cancer is divided into three categories.

- Stage IVA cancer has spread to a distant area or organ from the colon, such as the liver, lung, ovary, or a distant lymph node.
- Stage IVB cancer has spread to more than one distant area or organ from the colon.
- Stage IVC cancer has spread to the tissue that lines the wall of the abdomen and may have spread to distant areas or organs.

Similarly, stage IVA rectal cancer has spread a distant area or organ from the rectum, such as the liver, lung, prostate, or distant lymph node, stage IVB rectal cancer has spread to more than one distant area or organ and stage IVC rectal cancer has spread to the tissue that lines the wall of the abdomen and possibly other organs such as the liver, lungs, and brain.

Stage IV colorectal cancer follows a specific treatment plan. Firstly, surgery is performed to remove or reduce the size of the cancer in the colon, the rectum, or other organs of metastasis, such as the liver, lungs, prostate, or ovaries. After that, radiation or chemotherapy are implemented to relieve symptoms and enhance cancer cell destruction. Immunotherapy, which is a class of cancer drugs based on biologics that find and destroy colorectal cancer cells. Lastly, immunotherapy has recently emerged as a promising treatment option for this stage, particularly in patients with tumors that exhibit high microsatellite instability (MSI-H) or mismatch repair deficiency (dMMR), where it can lead to durable responses and, in some cases, long-term remission [18].

2.3 Liver metastasis mechanism and relevance

2.3.1 Metastasis definition and explanation

As suggested by the National Cancer Institute's official website, metastasis is the process by which cancer cells are transferred to other parts of the body. Metastatic cancers spread

from the original region to a distant part of the body and are usually referred to as "stage 4" cancers of their relevant type.

Lab experiments have shown that metastatic cancer cells have identical features with the primary cancer cells. Doctors can easily identify the metastatic cells in the foreign region and proceed with the appropriate treatment for the primary cancer they originated from. For example, colorectal liver metastatic cancer is treated like stage 4 colorectal cancer and not like liver cancer. CRC can spread to other parts of the body through the blood or lymphatic system.

2.3.2 Colorectal cancer statistics and prognosis

The Surveillance, Epidemiology, and End Results (SEER) Program collects and publishes data from population-based cancer databases, which are used to improve research, raise awareness and support efforts to reduce the impact of the disease across the U.S. population. It is managed and supported by the National Cancer Institute (NCI).

According to their published statistics, colorectal cancer represents 7.6% of all new cancer cases in the U.S., with an estimated 154,270 new cases and 52,900 deaths for 2025. The disease is more common in men than in women. The incidence rate was 37.1 per 100,000 men and women per year based on 2018–2022 data. Death rates increase with age with the most diagnosed group being individuals aged 65–74. Colorectal cancer is the second leading cause of cancer death in the United States, with a death rate of 12.9 per 100,000 men and women per year based on 2019–2023 data.

The stage of diagnosis plays a crucial role in determining treatment options and survival outcomes. When colorectal cancer is found early and remains localized (stage I) the prognosis is significantly better. Approximately 34.2% of cases are diagnosed at the local stage, with a 5-year relative survival rate of 91.5%. Figure 2.1 confirms the importance of early diagnosis, by showing the increased survival rates for patients diagnosed in stage I, relative to later stages. Moreover, it showcases the impact of diagnosis during metastasis, emphasizing the need for accurate diagnosis for metastatic lesions.

In contrast, metastatic colorectal cancer (mCRC) has a significantly worse outlook. Among people diagnosed with mCRC, approximately 70% to 75% survive beyond 1 year, 30% to 35% beyond 3 years, and fewer than 20% survive beyond 5 years from diagnosis [19]. Overall, the 5-year survival rate for mCRC is around 14% [20].

The liver is the most frequent site of metastasis, which in turn is the primary cause of death among CRC patients. About 50% of patients will develop liver metastasis at some point after diagnosis, while in 15–25% of cases, liver metastasis has already occurred upon diagnosis [21].

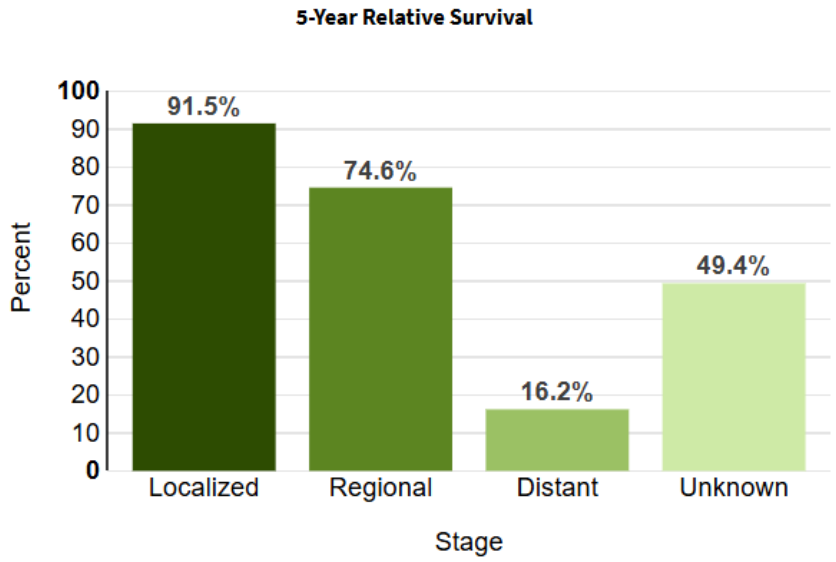

Figure 2.1: Survival rates by stage of diagnosis from SEER [1]

All the above statistics explain the current situation regarding the disease and the importance of further research to enhance diagnosis and treatment procedures.

2.3.3 Common sites of metastasis

As already stated, the liver is the most common site of metastasis in patients with CRC, due to the blood supply that exists between the large intestine and the liver. The anatomical and vascular connection from the liver to the colon happens through the portal venous system, a network of veins that carries blood from the abdominal portion of the digestive tract, spleen, pancreas, and gallbladder to the liver [13]. Up to 50% of CRC patients will develop liver metastases during their disease course, and the liver remains the leading cause of death among them.

Other common sites of metastasis include the lungs, bones, brain, or spinal cord. The finding of cancerous cells in these regions, especially after a patient has undergone treatment for CRC, is a strong indication of metastasis. Metastatic colorectal cancer is usually found after treatment at the original area of occurrence, when colorectal cancer cells are found in different regions of the body. This is different from recurrent colorectal cancer, which refers to the return of cancer at the original area or nearby lymph nodes after a period of remission, rather than in new parts of the body.

2.3.4 Mechanisms of metastatic spread to the liver

Colorectal liver metastasis occurs through a complex, multi-step process known as the invasion-metastasis cascade [20]. Cancer cells separate from their original location and enter surrounding tissue by rupturing barriers such as the basement membrane. After that, they move to distant organs like the liver via the lymphatic or circulatory systems. To survive in circulation, they use immune cells to form protective clusters.

Few of these cells can survive and proliferate once they arrive in a new organ. Many go into a dormant state, remaining inactive for extended periods before they may reactivate

and develop into metastatic tumors. Certain cancer cell subtypes, known as "metastasis-initiating cells," possess unique characteristics that enable them to reproduce and spread to new locations. These cells are supported by a favorable local environment called the "tumor microenvironment" and frequently develop from cancer stem-like cells, which can self-renew, differentiate, and initiate tumors [22].

The epithelial-mesenchymal transition (EMT), in which tumor cells lose cell-to-cell adhesion and acquire mobility and invasive capabilities, is another important factor that facilitates tumor metastasis. This process enables them to enter lymphatic or blood vessels, remain in circulation, and colonize distant locations of the body.

Additionally, cancer stem cells (CSCs) are essential for metastasis, since once they reach secondary organs, these cells can start the growth of new tumors due to their high capacity for self-renewal. Both mobility and colonization potential are promoted by the frequent overlap of EMT and CSC traits [22]. Lastly, genetic mutations, such as those in the *p53* gene, can also promote tumor spread and resistance to treatments [23].

2.3.5 Biological behavior of liver metastases

Most CRC metastatic cells reach the liver due to the portal venous system, which transports blood directly from the colon to the liver. Cancer cells engage with specialized cells in the liver area, including Kupffer cells, liver sinusoidal endothelial cells (LSECs), and hepatic stellate cells. The metastatic process is completed through these interactions, as they promote tumor survival, stimulate angiogenesis, and enable immune evasion [21]. Figure 2.2 visualizes the aforementioned cells of the hepatic region for better understanding of the metastatic process.

The metastatic progression in the liver typically follows four key phases. During the microvascular phase, cancer cells become lodged in sinusoidal vessels. This is followed by an extravascular, pre-angiogenic phase, then an angiogenic phase that provides essential oxygen and nutrients. Lastly, metastatic cells proliferate into detectable tumors during the growth phase. At each step, dynamic interactions occur between the invading cancer cells and various liver-resident or recruited cell types.

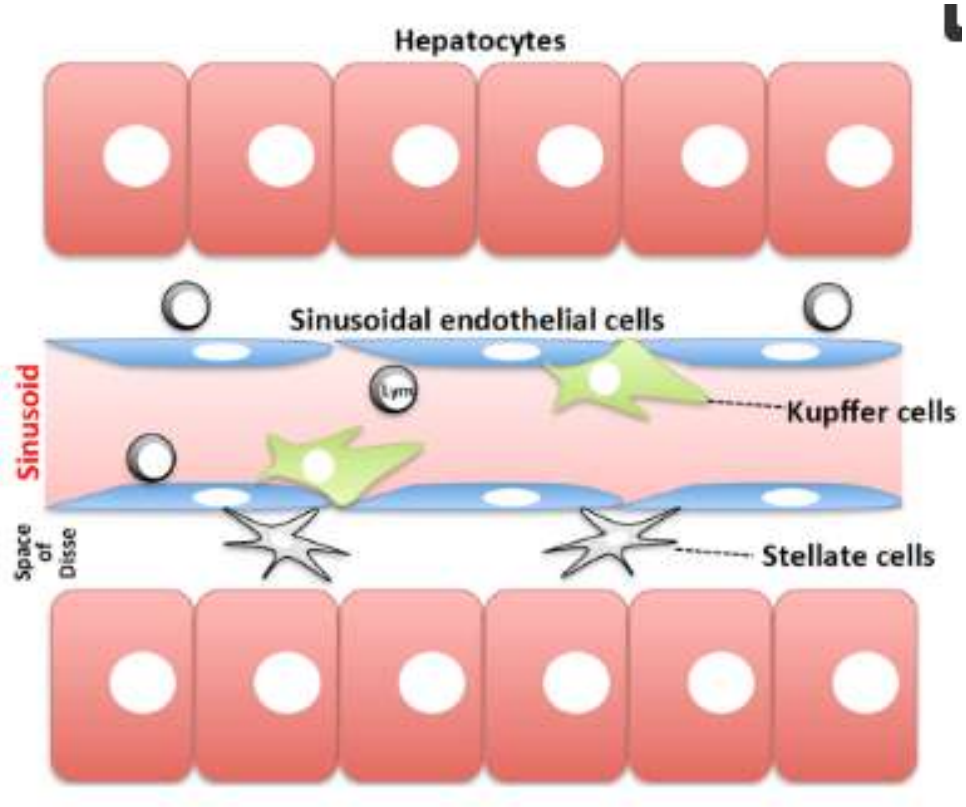

Figure 2.2: Schematic structure of the liver [2]

Once lodged in the liver, metastatic CRC cells encounter a distinct microenvironment composed of hepatocytes, LSECs, Kupffer cells, hepatic stellate cells, dendritic cells, and natural killer (NK) cells. These cell types have specialized roles in metabolism and immune modulation and are responsible for responding to antigens entering the liver via the portal circulation [21].

2.4 Diagnosis and imaging modalities

2.4.1 Role of imaging in diagnosis and monitoring

A health screening test is a medical test or procedure performed on asymptomatic patients to determine their likelihood of having a particular disease. Imaging-based screening enables the early detection of pathologic conditions before symptom appearance or physical examination findings. These tests are performed and analyzed by radiologists, who aim to decrease false-positive findings, successfully distinguish aggressive malignancies from benign ones, avoid over-treatment, decrease the radiation dose needed for screening modalities and establish best practices for managing pathologic findings [24].

Imaging-based screening tests are crucial for the detection, staging, and surveillance of CRC and CRLM. Early detection through high-quality screening and diagnostic imaging significantly reduces CRC incidence and mortality rates [25]. Colonoscopy remains the main diagnostic medical procedure, supported by quality metrics such as adenoma detection rates that can help identify cancer risk [26]. Advanced imaging modalities, like computed tomography (CT), magnetic resonance imaging (MRI), and positron emission tomography (PET) scan, offer precise visualization of both primary tumors and

metastatic sites, crucial for treatment planning. AI-enhanced technologies have further improved diagnostic accuracy and efficiency, particularly in non-invasive techniques like CT colonography and capsule endoscopy [27].

2.4.2 Clinical characteristics of screening methods and AI influence

A range of effective screening methods has been developed to detect abnormal tissue that could belong to a premalignant lesion or an early-stage tumor. These include invasive techniques, like colonoscopy and flexible sigmoidoscopy and more passive options, like capsule endoscopy and imaging exams like CTs and MRIs. Additionally, there are stool and blood-based tests, including the guaiac fecal occult blood test (FOBT), fecal immunochemical test (FIT), and multitarget stool DNA (mt-sDNA) test, that are used to detect cancerous cells [25].

Virtual colonoscopy, or computed tomographic colonography (CTC), offers a non-invasive alternative and benefits from AI models that enhance image analysis, enabling better discrimination of neoplastic and non-neoplastic lesions. Capsule endoscopy (CE) serves as an alternative for incomplete colonoscopy cases and can also benefit from AI tools for automated polyp detection, reduced human error and review time [28].

Blood-based screening methods are also emerging, with the help of AI-assisted models using blood tests and electronic health data that can assess CRC risk with high sensitivity and specificity. These models can perform thorough analysis of complete blood count data, serum biomarkers, and circulating tumor cells [28]. Sigmoidoscopy examines only the distal colon, has >95% sensitivity for CRC in that region, and is less invasive but misses proximal lesions and may be uncomfortable without sedation [15].

2.4.3 CRLM Imaging – prognosis and challenges

The main prognostic indicators for CRLM lesions include number of metastasis, size and location of the lesions, response to systemic chemotherapy, time of diagnosis for different occurrences (e.g., synchronous vs. metachronous), the presence of tumor cells within blood or lymphatic vessels, as well as the quality of the underlying liver tissue [29].

Imaging techniques for CRLM entail notable challenges that impact both detection and treatment planning. While CT remains the most available and quick modality, its sensitivity drops significantly for lesions smaller than 1 cm, particularly after chemotherapy [30]. PET/CT is great at identifying extrahepatic disease but struggles to detect small or mucinous metastasis, limiting its utility for liver lesions, while integrative PET/MRI may improve lesion detection and confidence, but is a technically complex procedure.

Some specialized MRI techniques achieve higher sensitivity per lesion; however, it is a time-consuming and costly procedure, with significant error margins due to patient movement. Advanced imaging techniques, including radiomics and AI-driven analysis, show promise in enhancing detection and response assessment, while there is still room for improvement for small sample sizes, workflow integration, and the need for validation protocols [31].

Colorectal liver metastasis displays some distinct imaging characteristics that help doctors differentiate them from other liver lesions. One of the most common features is the presence of a peripheral rim, especially visible on contrast-enhanced imaging, caused

by central necrosis and peripheral viable tumor tissue, which enhances during the portal venous phase. These lesions are typically hypo-vascular, appearing less enhanced on CT or MRI scans, than the liver parenchyma that surrounds them. On MRI scans, CRLM often appear brighter than the surrounding liver tissue on T2-weighted images, which helps in identifying tumors. This brightness is due to the water content and structural changes within the tumor. In addition, diffusion-weighted imaging (DWI), a type of MRI that detects how water molecules move within tissues, shows restricted diffusion in CRLM. On some instances, the liver surface appears pulled inwards due to the presence of a metastasis, creating a "capsular retraction" feature. Additionally, CRLM often demonstrate rapid growth and irregular margins, and may show satellite lesions nearby [31].

2.5 Primary treatment methods and challenges

2.5.1 Clinical implications and treatment

Treatment for CRC usually involves multiple steps, starting with surgical resection when the disease is localized and adding chemotherapy and radiation for advanced stages or high-risk cases. Systemic therapies, such as combinations of cytotoxic medications, targeted biological agents, and immunotherapies, are the standard treatment for metastatic colorectal cancer. Significant clinical challenges still exist despite advancements in these treatments. These include addressing tumor heterogeneity and resistance mechanisms, optimizing treatment regimens, finding trustworthy biomarkers for tailored therapy, and making sure genomic testing that analyzes the patient's complete set of genes, is available to all populations. While the number of options is constantly growing due to ongoing research into new targeted agents and immunotherapeutic approaches, improving outcomes for CRC patients still primarily depends on increased research and accessibility of methods [32].

2.5.2 CRLM treatment options

While liver metastasis can be quite challenging, liver-directed therapies offer important options for CRLM, especially when surgical resection is not possible or safe. While curative resection is possible in only 10–20% of cases, local treatments help manage tumors and sometimes enable surgery by reducing tumor burden. Most common treatment options aiming to reduce or even remove metastatic tumors include chemotherapy, HAI therapy, ablation, radiation, cryotherapy or heat. Apart from the tumor region, the nearby tissue is usually unaffected by these less invasive liver-focused methods. The major treatment options for CRLM as described by the Colorectal Cancer Alliance are presented below.

During hepatic artery infusion, concentrated chemotherapy is directly delivered to liver tumors via a pump connected to the hepatic artery. This approach reduces systemic exposure and enhances local tumor control. It may enable resection for previously unresectable liver metastasis and is also used to decrease recurrence rates after surgery. The major side effect from HAI is liver toxicity, so liver enzymes should be monitored closely after treatment [30].

Other common measures for CRLM are embolization techniques, which block blood flow to tumors, depriving them of oxygen and nutrients vital for their growth. More specifically, portal vein embolization is used to promote growth and hypertrophy of the

healthy liver remnant, increasing the safety and success rate of future surgery. Though effective, it carries a risk of post-embolization syndrome, including pain, fatigue, nausea and fever [31].

Selective Internal Radiation Therapy (SIRT) introduces tiny radioactive spheres into the liver's arterial supply. A small flexible tube is guided through an artery into the liver and the microspheres are delivered directly into the tumor, where they release radiation. This targeted therapy delivers a dose of internal radiation up to 40 times higher than conventional radiation therapy. It is typically used in patients who have failed or are ineligible for standard chemotherapy. It offers better local control and delayed disease progression, though survival benefit is still being researched [33]. Side effects include abdominal pain, nausea, loss of appetite, mild fever and increased fatigue, however the procedure is quick and painless.

Another radiation alternative is stereotactic body radiation therapy (SBRT) [34]. It enables highly precise radiation delivery over just a few sessions, with success rates up to 90% in select patients in contrast to standard radiation methods that fluctuate around 40%. It is especially valuable for oligometastatic cases or when other ablative methods are discouraged. Despite its greater dosage, SBRT has relatively few side effects, mainly short-term fatigue.

Microwave Ablation (MWA) uses microwave energy to heat and destroy liver tumors, often guided by imaging such as CT or ultrasound. An ablation antenna into the center of the liver tumor, where it delivers thermal energy to destroy cancer cells. It is a rarer procedure, best suited for tumors less than 3 cm and in locations difficult to reach surgically [35]. Recovery is quick in laparoscopic cases, but deeper lesions or open procedures may increase risk of complications.

Radiofrequency Ablation (RFA) applies high-frequency electrical currents to generate heat and destroy cancer cells. The procedure is usually done by inserting a needle through the skin, then placing a probe through the needle and positioning it in the liver tumor. Alternatively, it can be done laparoscopically or even with open surgery. It is a widely used non-surgical approach, either as a standalone treatment or to complement surgery in cases hardly resectable [36]. Most patients recover quickly, although complications like skin burns or infection are possible.

2.5.3 Specialized treatment options for mCRC

For cases of metastatic colorectal cancer (mCRC) that cannot be resected, systemic therapy remains the primary treatment approach. This includes chemotherapy, biologic agents such as antibodies targeting growth factors, immunotherapies, or a combination of these strategies.

Approximately 50% of patients with metastatic CRC have tumors that are "wild-type" for KRAS, NRAS, and BRAF genes, meaning the genes are not mutated in the tumor cells. These genes greatly affect cell growth and division, therefore when mutated, they can drive cancer growth and worsen treatment options. Wild-type tumors, particularly for these genes, are generally more responsive to certain targeted antibody therapies, which can increase median survival by 2 to 4 months compared to chemotherapy alone. Progress in molecular profiling has improved the ability to tailor treatments to the biological characteristics of individual tumors. Although complete cures in metastatic CRC are still rare, personalized therapies have been increasing patients' life span [37].

The spread of CRC to distant organs is the leading cause of death and continues

to present major treatment challenges. However, substantial advances have been made in the development of targeted therapies, with several ongoing clinical trials and FDA-approved drugs being available to patients [20]. Immunotherapy has shown meaningful success in tumors with high microsatellite instability (MSI-H), marking a significant step forward in mCRC treatment [38].

In addition, several emerging technologies including immunostimulatory cytokines, nanotechnology, and the use of oncolytic viruses, bacteria, and therapeutic peptides exhibit promising results in other cancers and hold potential future applications in mCRC as well.

2.6 Segmentation in Tumor Representation

2.6.1 Role and challenges of segmentation in diagnosis

Diagnosis and treatment planning for CRLM are heavily influenced by the recent advancements in AI and Deep Learning technologies. Deep Neural Networks have been found to be especially effective at image classification and segmentation tasks which are a key part of medical imaging methods for improved diagnostic procedures [39].

In medical imaging, 3D tumor segmentation refers to the process of finding and outlining the exact boundaries of a tumor within volumetric image data like CT scans. This task is crucial for accurate diagnosis, treatment planning, and monitoring of tumor progression or response to therapy.

In the case of metastatic liver tumors, segmentation tasks are quite challenging, due to the variety in tumor shape, size, and appearance, as well as the lack of contrast between malignant tissue and surrounding healthy liver. Nevertheless, accurate segmentation is necessary, as it provides the essential foundation for radiomic analysis and guides the physician's decision making [19].

2.6.2 AI Advancements and concerns in imaging and segmentation

Automated tumor segmentation has been recognized by clinicians as a tool that accelerates image analysis, and minimizes both oversights and human error—ultimately contributing to improved patient care [40]. Artificial intelligence (AI), particularly deep learning, has emerged as a promising approach to enhance both the precision and speed of image-based classification while keeping high standards of clinical quality.

As digital transformation projects in healthcare keep expanding, adding decision-support systems with standard workflows can speed up adoption and allow analysis of multiple tumor samples at once. This approach could help tackle tumor heterogeneity and improve prognosis accuracy [41].

However, the lack of transparency in how deep learning models make decisions remains a major obstacle to building trust and gaining widespread clinical adoption [42]. To overcome these barriers, there is a growing consensus in the medical community on the need for greater transparency, the validation of techniques across diverse patient populations, and the establishment of clear regulatory frameworks.

In the next chapter we are going to expand on the foundational concepts of AI and deep neural network theory that set the ground for the later experiments on CRLM 3D segmentation.

Chapter 3

Deep Learning Fundamentals

3.1 Basics of deep learning

3.1.1 Introduction to machine learning and deep learning

Machine learning (ML) refers to a set of methods that automatically determine patterns in data and then utilize them to predict future data or enable decision making under uncertain conditions. The most representative characteristic of ML is that it minimizes human interventions and mainly relies on data for the decision process. The program learns from the analysis of training data, and follows with a prediction upon new data is input [43].

Deep learning (DL) is one of the fastest growing branches of artificial intelligence in recent years. The scientific community has focused on DL because of its versatility, high performance, and high generalization ability, among many other qualities. In addition, the development of more advanced computers along with the increased availability of medical data has also increased interest in DL applications for medicine [44]. Other studies, also emphasize the excellent performance of DL in detection, classification and segmentation tasks for medical images, with results comparable to medical professionals [45]. Therefore, it is evident that the presentation and explanation of key ML and DL concepts is necessary for a successful medical image segmentation project.

3.1.2 Types of DL tasks

Deep learning techniques can be utilized in a variety of tasks, each applied for unique objectives in domains like robotics, manufacturing, medical imaging, text detection etc.

Regarding medical imaging applications, these are the primary categories where DL is applied:, based on [46]:

- **Classification**: The goal is to match each input image (e.g., CT or MRI slice) to a specific category, such as determining between benign and malignant tumors.
- **Segmentation**: Produces labels of pixels or voxels that best outline the shape of the input object. For example, segmentation is used for the precise analysis and mapping of anatomical and pathological regions of whole organs or tumors.
- **Detection/Localization**: Involves the identification and localization of regions of interest, usually by drawing a bounding box around the specified object. For example it is used for the detection of nodules or lesions in diagnostic tasks.

- **Anomaly Detection**: Highlights patterns that differ from healthy baselines, useful for identifying rare or early-stage conditions in cases where labeled data is limited.

Apart from these tasks, there are other popular DL applications, which can be related to medical imaging but are mostly developed for different fields of study:

- **Generative Modeling**: GANs (Generative Adversarial Networks) are refer to the ML technique of utilizing two neural networks, a generator and a discriminator, which compete against each other to generate realistic synthetic data. These tasks usually include data augmentation, image synthesis, or domain adaptation.
- **Reinforcement Learning (RL)**: An agent is used to make decisions in a specified environment, based on sequential policies, aiming to maximize a cumulative reward. This technique is applied in robotics, autonomous driving, and adaptive treatment strategies.
- **Natural Language Processing (NLP) Tasks**: It refers to the processing of natural language information by a computer model, with the goal of understanding and generating human-like language. It is developed for text classification, translation, and summarization tasks.

3.1.3 Supervision categories

A key aspect of deep learning theory is supervision. The presence of labels in the training dataset, which dictate the correct categorization of the data, determines whether a deep learning project is supervised, unsupervised, or semi-supervised.

In **supervised learning**, all training images are accompanied by the corresponding "ground truth" labels (masks) to facilitate the model's optimization. For each testing image, the optimized model generates a likelihood score to predict its class. This prediction is based on the model's understanding of the relationship and structure of the input image-label pairs [47]. Supervised learning is the most usual training method for medical image segmentation tasks. The models are trained with a large number of annotated medical images in order to predict the segmentation masks of a foreign image sample [48].

In **unsupervised** learning, the model analyzes the patterns or hidden data structures of the input images without the help of labels, using statistical methods such as clustering algorithms and density estimation [47].

Lastly, if only a distinct part of training data has labels, the model uses them to grasp the basic patterns and is later enhanced by learning subtle and fine-grained features from the unlabeled data. This type of learning approach is defined as **semi-supervised** learning [49].

3.1.4 Transfer learning

Another fundamental ML technique for deep learning segmentation, designed to enhance model performance by utilizing information from already trained models on similar tasks, is transfer learning. [50].

This technique was developed to deal with corrupt or scarce annotations. With transfer learning, knowledge obtained from models trained on large datasets in irrelevant

domains or tasks, like ImageNet, is leveraged for medical image segmentation, often improving performance in cases with limited or poor label data [48]. It is a widely used technique, given the limited availability of sufficiently annotated medical image datasets, with the ultimate goal to initialize model weights, saving valuable time and computational units. The pretrained networks are adjusted to the specific needs of each medical task, in a process called fine-tuning.

In the context of this CRLM segmentation task, however, transfer learning has not been widely adopted due to the fully annotated dataset provided and the highly specific nature of the disease's imaging patterns, which differ considerably from those in general-purpose datasets. Transfer learning is still very significant in the clinical field for enhanced performance in situations when obtaining ground-truth labels is difficult, as shown in relevant research [51, 52].

3.2 Neural Networks: Key Concepts

3.2.1 Layers, Back-propagation and Activation

Neural networks are defined as algorithmic models composed of structured layers of interconnected neurons. They are designed based on the learning processes of the human brain. Through training on specific data they can gradually recognize patterns and generate predictions by adjusting connection weights [53]. They consist of three main layers:

- The input layer receives raw feature data, meaning that each neuron corresponds to one attribute of an input sample.
- The hidden layers can be multiple and are responsible for the main computation. They consist of of neurons that apply weighted transformations and non-linear activations to the incoming signal.
- The final (output) layer is responsible for generating the the final prediction, which can be either class labels, a continuous output, or probabilistic distributions such as softmax outputs in classification tasks

Data flows forward from input to output during forward propagation. The model's predictions are evaluated using a loss function, which calculates the difference between predicted and actual values. This loss is then minimized via backpropagation, where gradients of the loss with respect to model parameters are computed and used to update weights and biases, usually via optimization algorithms.

Another fundamental component of NNs is activation functions for both the hidden layer and the output layer. They are mathematical functions that introduces non-linearity into the model, allowing the network to learn and represent complex patterns in the data.They are applied to the output of a neuron. Without this feature a neural network would behave just like a linear regression model - which applies a a linear equation to examine the relationship between a dependent variable and one or more independent variables - no matter how many layers it has [54]. Some of the fundamental activation functions are the following:

The Rectified Linear Unit (ReLU) has become the state-of-the art activation function due to its simplicity and improved performance.

$$\mathrm{ReLU}(x) = \max(0, x)$$

While the standard ReLU function is widely used, several variations have been proposed to address its limitations, such as the "dying ReLU" problem, where neurons can become inactive and stop learning.

Leaky ReLU: The Leaky ReLU introduces a small, non-zero slope α for negative input values to allow a small gradient to flow even when $x < 0$. It is defined as:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{if } x < 0 \end{cases} \tag{3.1}$$

where α is a small constant (e.g., $\alpha = 0.01$).

Parametric ReLU (PReLU): The PReLU extends Leaky ReLU by making α a learnable parameter, allowing the network to learn the negative slope during training:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ ax & \text{if } x < 0 \end{cases} \tag{3.2}$$

where a is optimized through backpropagation.

Logistic Sigmoid and Tanh activation functions were used in earlier NN theory. The Logistic Sigmoid is a very popular and traditional nonlinear function. However, output can become saturated at very high or very low input values, which can cause the vanishing gradient problem. This occurs when the gradient of the objective function with respect to a parameter becomes extremely small, resulting in insignificant updates to the parameters during training with stochastic gradient descent. As a result, learning slows down dramatically, and in severe cases, training can effectively stall.

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}$$

$$\text{Tanh}(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

Lastly, the Softmax function is used in the output layer of a multi-class classification neural network to convert a vector of raw prediction scores (logits) into probabilities. It ensures that the resulting probabilities are non-negative and sum up to 1 across all classes.

For a given input vector $\mathbf{z} = [z_1, z_2, \ldots, z_K]$, where z_k represents the raw output (logit) corresponding to class k and K is the total number of classes, the Softmax function is defined as:

$$\text{Softmax}(z_k) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}. \tag{3.3}$$

Here, $\text{Softmax}(z_k)$ gives the probability that the input belongs to class k. The final output of the Softmax layer is a probability distribution over all classes. To make a class prediction, the model typically selects the class with the highest probability:

$$\hat{y} = \arg\max_{k} \ \text{Softmax}(z_k). \tag{3.4}$$

3.3 Evolution of neural networks in deep learning

3.3.1 Artificial neural networks

Artificial Neural Networks (ANNs) are computational models based on the human brain's processing functions. They are consist of interconnected units called perceptrons or artificial neurons, which receive and process inputs to generate an output. Each perceptron is activated and passes its signal to the next layer of the network, based on a mathematical activation function. These networks learn through supervised learning, adjusting their internal weights based on many training instances. Once trained, ANNs can generate automatic, accurate responses to new relevant inputs [44].

A multi-layer perceptron (MLP) is a type of ANN consisting of multiple layers of neurons, as shown in 3.1. The neurons in the MLP typically use nonlinear activation functions, allowing the network to learn complex patterns in data. ML applications highly rely on MLPs, due to their ability to learn nonlinear relationships in data, simplifying tasks such as classification, regression, and pattern recognition.

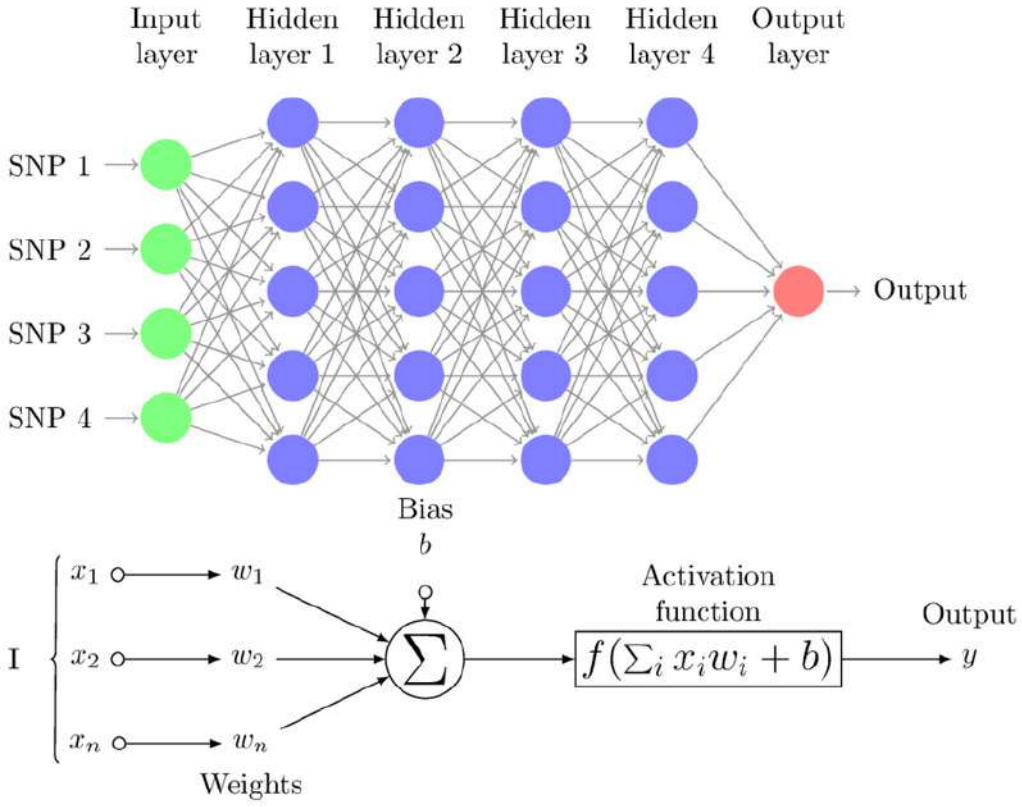

Figure 3.1: Example of MLP [3]

Deep neural networks (DNNs) are the extension of conventional artificial neural networks. The main difference is the number of hidden layers, which is much greater in DNNs, than the shallower normal NNs. There is a wide range of deep learning architectures, including fully connected deep neural networks (DNNs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their variants such as long short-term memory networks (LSTMs). As the size and the layer count of a network increases, it becomes more complex and requires more time and resources for training [39].

Deep neural networks (DNNs) have significantly improved diagnosis, treatment planning, and patient care in medical applications. With large medical datasets as input, they develop the ability to capture significant characteristics and patterns from medical images, resulting in more accurate and effective analysis.

These types of networks work best with GPU-based architectures that require significantly less training time than classical CPUs [54]. Originally created for graphics rendering, modern GPUs are massively parallel processors that are excellent at speeding up deep neural network (DNN) training because of their capacity to effectively perform thousands of simultaneous operations. When compared to CPUs, they significantly cut down on training time, allowing complicated models to be trained in a matter of hours or days [55].

3.3.2 Limitations of shallow networks & growth of DNNs

Although the concept of ANNs was first presented in the 1950s, its applications for actual problems were severely limited because of issues with overfitting and vanishing gradients. These resulted in a lack of processing power, difficulty in deep architecture training,and most importantly, a lack of sufficient training data for the system. It is very likely that a network with an excessive number of nodes and hidden layers will eventually learn every training pattern in the training data a phenomenon known as overfitting.

The vanishing gradient problem, which arises when the gradients of the loss function with respect to the weights of the early layers become increasingly small, is one of the main obstacles in DNN training. The result is slow convergence or even stagnation, as early layers receive little to no updated weight information during backpropagation. The selection of activation functions and optimization techniques in DNNs is primarily responsible for the vanishing gradient issue [56].

However, with the evolution of big data, GPUs, and novel training algorithms training algorithms, many obstacles have now been overcome. In a variety of domains, including medical imaging, these deep learning techniques have demonstrated remarkable results in simulating human behavior [44]

3.4 Convolutional Neural Networks (CNNs)

3.4.1 Convolutional neural network basics

Convolutional Neural Networks (CNNs) are comprised of neurons that improve themselves through learning, similar to normal ANNs. After receiving the input and activation function, the network continues to express score function (the weight) from the input raw image vectors to the final output of the class score. The final layer will include loss functions related to the classes.

As described by [4], CNNs primarily focus on images used as inputs. One key distinction of CNNs lies in their neurons' arrangement. Instead of a simple one-dimensional structure, CNN layers process data in three dimensions: height, width, and depth, where depth refers to the number of feature maps (or channels) in an activation volume. Unlike fully connected ANNs, each neuron in a CNN layer connects only to a small, localized region of the previous layer.

For example, an input image of size of $64 \times 64 \times 3$ (height, width, RGB channels) might be transformed into a final activation volume of size $1 \times 1 \times n$, where n being

the number of output classes. This efficiently reduces the original input's rich spatial information into class scores across the depth dimension.

3.4.2 CNN structure

A typical CNN architecture consists of three main types of layers: convolutional layers (for feature extraction), pooling layers (for downsampling and spatial reduction), and fully connected layers (for final decision-making). By stacking these layers in order, CNNs can progressively learn and combine simple patterns into complex, high-level features for classification or other vision tasks.

Therefore, the workflow of the CNN architecture, visualized in 3.2 begins with the pixel values of the image being held by the input layer.

After that, the convolutional layer passes small, learnable filters (kernels) across the input's spatial dimensions, to processes local regions. At each position, it computes a scalar product between the filter weights and the corresponding input values, resulting in a 2D activation map. Each kernel is designed to learn specific features, like edges or structures and is activated when that feature appears in the input. As the network gets deeper the appropriate kernels are applied to capture more complex information. The full output volume of the layer consist of all the activation maps stacked along the depth dimension. Activation functions, such as the ReLU, are then applied to introduce non-linearity. Convolutional layers are controlled by key hyperparameters, such as depth, stride, and zero-padding, which determine the preservation of spatial information and the overall model's complexity.

The pooling layer operates over each activation map in the input and scales its dimensionality using the "MAX" function, reducing the number of parameters within that activation. Lastly, the fully-connected layers attempt to produce accurate class scores from the activations, to be used for classification. It is also usual practice for ReLU to be used between these layers, to improve overall performance.

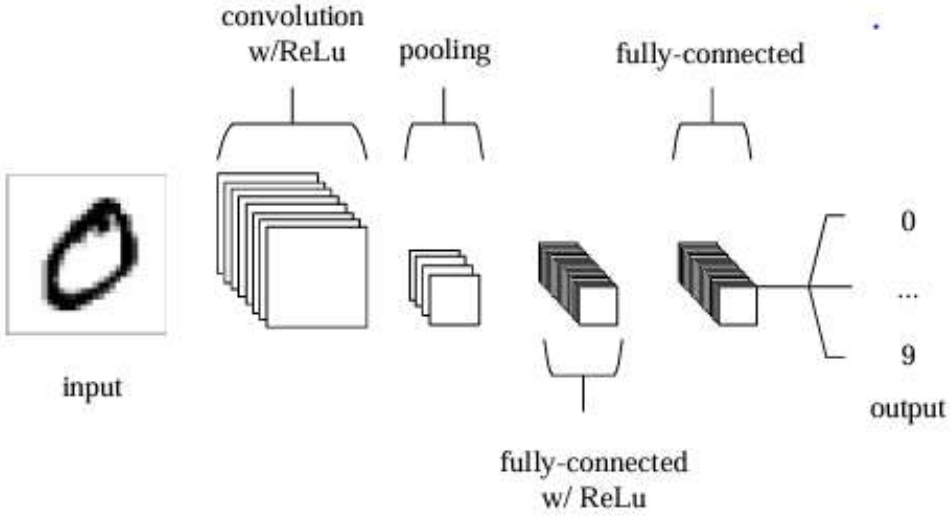

Figure 3.2: CNN architecture [4]

[57] introduced a variation of CNNs, the Fully Convolutional Network (FCN), in which the final fully connected layers are replaced with convolutional layers. This design preserves spatial information, enabling dense pixel-wise predictions over the entire image.

FCNs avoid the drawbacks of patch-wise prediction by combining high-resolution activation maps with upsampled coarse outputs to enhance localization accuracy and process an entire image in a single forward pass [58].

3.4.3 Role in medical imaging

It is evident that CNNs, being specifically designed for image inputs, have become a standard approach for medical imaging tasks, such as disease detection, tumor localization, and diagnosis [59]. These models have significantly reduced the need for manual feature extraction, with their ability to automatically understand hierarchical features from images.

CNNs have been improved through model ensembling, architectural optimizations, and the addition of interpretable techniques that have enhanced their interpretation of classification tasks. They have enabled methodical and extremely accurate medical image classification, particularly when paired with cautious and adequate data preprocessing [60].

The most fundamental architecture used in modern image segmentation tasks, the U-Net, is developed based on the foundations of CNNs. While standard CNNs were originally designed for classification tasks they struggled with pixel-level localization, which is crucial in biomedical imaging. The U-Net utilized the concept of fully convolutional networks (FCNs) to design a symmetric encoder-decoder structure that is going to be analyzed in later chapters. The result was a great increase in segmentation accuracy, even with very limited training data, which is often the case in biomedical applications [6].

Beyond U-Net, several other CNN-based architectures have advanced segmentation. SegNet focuses on efficient upsampling with encoder–decoder designs. DeepLab uses atrous convolutions and conditional random fields for sharper boundaries. Also variations of U-Net were developed, like Attention U-Net, which enhances feature selection through attention mechanisms.

3.5 Detection and evaluation metrics and loss functions

3.5.1 Metrics for segmentation

In segmentation tasks the main objective is to measure the volumetric overlap between the original and the predicted labels. For this purpose the primary metrics used for these tasks are Dice Score, Surface Distance and Volume Similarity [61].

The Dice score assesses the level of overlap between the predicted and ground truth segmentation masks. For example, given two binary masks X and Y, the formula is:

$$\mathrm{DSC}(X,Y) = \frac{2 \times |X \cap Y|}{|X| + |Y|} = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Surface distance metrics are correlated measures of the distance between the surfaces of the input and the predicted region. For this purpose, the Euclidean distance is applied to determine the shortest distance of an arbitrary voxel u to a set of surface voxels $S(X)$ as:

$$d(v, S(X)) = \min_{s \in S(A)} \|v - s\|$$

Based on this distance, the average symmetric surface distance (ASSD) is then given by:

$$\text{ASSD}(X, Y) = \frac{1}{|S(X)| + |S(Y)|} \left(\sum_{x \in S(X)} d(x, S(Y)) + \sum_{y \in S(Y)} d(y, S(X)) \right)$$

Lastly, the relative volume difference (RVD) measures the volume difference directly, disregarding the overlap between ground truth and the prediction, as follows:

$$\text{RVD}(X, Y) = \frac{|X| - |Y|}{|Y|}$$

3.5.2 Metrics for detection

For detection and classification tasks, where the main objective is to predict the target variable, additional metrics are introduced for detailed model evaluation. These metrics are calculated globally and require a certain correspondence between predicted and ground truth labels [61].

To begin with, a fundamental metric for classification models is Accuracy. It provides a quick estimation of the correctness of the model's predictions. It is calculated as the ratio of correct predictions to the total number of input samples. Its optimal usage is done when each class has an equal number of samples.

$$\text{Accuracy} = \frac{\text{\# correct predictions}}{\text{Total number of input samples}}$$

In segmentation tasks, the concepts of true positives, false positives, true negatives, and false negatives are applied at the pixel (or voxel) level rather than at the image level, as in standard classification. A true positive (TP) When a pixel is correctly identified as being part of the region of interest, it is deemed as a true positive (TP). A true negative (TN) is a pixel correctly identified as background or outside of the specified region of interest. A false positive (FP) occurs when background pixels are incorrectly classified as positive pixels for the requested region, while a false negative (FN) refers to pixels that are highlighted in the ground truth mask, but the model fails to detect and instead classifies as background. The detection metrics presented below are based on different relationships between these definitions and aim to provide a more complete evaluation of model performance [61].

Precision is a measure of a model's performance that shows its ability to correctly identify the actual positive samples.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall describes the ratio of correctly predicted positive instances to the total actual positive instances. It determines the accuracy of the positive predictions made by the model.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1-Score is the harmonic mean of precision and recall, ranging from 0 to 1. It balances the trade-off between correctly identifying positive cases and not missing relevant ones. A higher F1-Score indicates better overall performance of the model.

$$\mathrm{F1} = \frac{2}{\mathrm{Precision}^{-1} + \mathrm{Recall}^{-1}}$$

All the above metrics are crucial to evaluate the model's performance and pinpoint any areas that require improvement in the model's architecture and training process.

3.6 Deep network training

3.6.1 Overfitting and data augmentation

The ultimate goal of training a DL model is to learn patterns that capture the basic features of the data rather than memorizing specific training samples. A well-trained model should maintain high performance in unseen test data, in addition to performing well in the training set. This ability to generalize is critical in applications such as medical imaging, where models are expected to perform reliably across various patients, medical equipment, and data acquisition methods.

As suggested by [62], one of the most common obstacles for successful generalization is overfitting. Overfitting refers to the adoption of patterns that are too specific to the training data, including noise or irrelevant details, which can results in poor performance on unseen data. It can be observed by comparing training and validation performance over time and especially the corresponding error trends. If the training error continues to decrease, while the validation error may begin to rise at certain point, it strongly indicates that the model is just memorizing the training data rather than capturing meaningful general features. In contrast, a model with good generalization displays decreasing trends for both training and validation errors.

Data augmentation is a strategy developed to tackle overfitting issues. It increases the diversity of the training set by applying transformations such as rotations, flips, scaling, cropping, or noise to existing training samples. These transformations enhance the diversity of the input data, helping the model become resistant to irrelevant changes in data structure while focusing on the essential features needed for accurate predictions.

3.6.2 Normalization techniques

Normalization is another common technique applied in DL models with the purpose to accelerate and stabilize the training process. Data normalization ensures that features are all scaled to similar ranges, preventing certain variables from dominating the learning process. Batch Normalization (BN) is the most prominent normalization technique. It ultimately reduces overfitting and fastens model convergence by computing the mean and variance of the inputs to each layer within a mini-batch of data. Based on this on this, numerous other techniques have been developed, like instance, layer, group and positional normalization [63].

Other simpler techniques include min-max scaling or linear normalization which maps data to a specific range to reduce extreme deviations and z-score normalization, which uses mean and standard deviation to produce normalized values from unstructured data. All these techniques play a crucial role in improving the stability, the efficiency and the

generalization ability of deep networks. More specific techniques for medical imaging tasks are analyzed in later chapters.

3.6.3 Regularization techniques

Regularization refers to another collection of strategies designed to prevent overfitting. Their main purpose is to improve the model's ability to generalize while keeping training efficient. The most widely adopted methods include:

- **L1 & L2 Regularization:** L1 (Lasso) regularization adds a penalty based on the absolute value of weights, enhancing sparsity, as many weights become zero. L2 (Ridge) regularization encourages smaller, more evenly distributed weights by adding a penalty based on the square of the coefficients, shrinking them towards zero. This technique makes decision boundaries smoother and improves generalization. The most common implementation of L2 for optimization algorithms is weight decay. The optimizer reduces the weights at each update step, preventing parameters from growing too large and improving generalization.
- **Dropout:** Dropout is another regularization technique, that randomly deactivates a subset of neurons and their connections during training [64]. With this method, co-adaptation of neurons is prevented and the model trains a collection of "thinned" networks containing the units that remained active, which ultimately reduces overfitting and improves robustness.
- **Early Stopping:** Another popular technique is early stopping, which involves stopping the training process when the validation loss stops improving. By restricting the number of iterations the model is trained for, early stopping can successfully lower the chance that the training data will be memorized. However, lower patience values can sometimes result in premature stopping preventing the model from reaching its full potential. On the other hand, longer training times do not necessarily improve validation accuracy, and early stopping can help prevent overfitting and shorten training time and cost. Therefore, the training epochs and patience selected must be carefully considered as they can strongly impact the final training results [65].

3.6.4 Loss functions

In DL tasks, loss functions are used to quantify the difference between predicted outputs and the actual ground truth, essentially guiding the optimization process. Their value is the main indication that the model's prediction accuracy is improved, therefore the minimization of the loss becomes the main target of the training process. Each task category best works with different loss functions. For example, binary cross-entropy and categorical cross-entropy are conventional in classification, whereas mean squared error (MSE) and mean absolute error (MAE) are commonly employed in regression. Custom losses like Dice Loss or Hinge Loss are frequently used in more specialized tasks like segmentation or object detection in order to capture more specific requirements. A key component of efficient deep learning training pipelines is the careful selection of a loss function, which has a direct impact on model convergence, resilience, and generalization to unknown data [66]. Some of the most common loss functions as presented by [67] are discussed below.

Cross-Entropy loss

Cross-entropy loss is one of the most widely used loss functions in deep learning for classification and segmentation tasks. It measures the difference between the predicted probability distribution $\hat{y}_i$ and the true label y_i. For a single voxel, it is defined as:

$$\mathcal{L}_{CE}(\hat{y}_i, y_i) = \begin{cases} -\log(\hat{y}_i), & \text{if } y_i = 1 \\ -\log(1 - \hat{y}_i), & \text{if } y_i = 0 \end{cases} \tag{3.5}$$

The total loss is computed as the mean across all voxels in the image volume, with each voxel contributing equally.

Focal loss

Focal loss reduces the weight for easy examples and focuses training on harder and misclassified ones, in an effort to address class imbalance issues. It modifies cross-entropy with a modulating factor γ and a weighting factor α, resulting to this form:

$$\mathcal{L}_{FL}(\hat{y}_i, y_i) = \begin{cases} -\alpha(1 - \hat{y}_i)^\gamma \log(\hat{y}_i), & \text{if } y_i = 1 \\ -(1 - \alpha)\hat{y}_i^\gamma \log(1 - \hat{y}_i), & \text{if } y_i = 0 \end{cases} \tag{3.6}$$

where $\alpha \in [0, 1]$ balances the importance of positive and negative samples, and $\gamma \geq 0$ reduces the relative loss for well-classified examples.

Dice loss

Dice loss is derived from the Dice Similarity Coefficient (DSC), commonly used in segmentation tasks to measure the overlap between predicted segmentation $\hat{Y}$ and ground truth Y. The binary form, in cases where the background weight is set to 0 is defined as:

$$\mathcal{L}_{DSC}(\hat{Y}, Y) = 1 - \frac{2\sum_{i=1}^{N} \hat{y}_i y_i + \epsilon}{\sum_{i=1}^{N} \hat{y}_i + \sum_{i=1}^{N} y_i + \epsilon} \tag{3.7}$$

where N is the total number of voxels, and ϵ is a small smoothing constant to prevent division by zero. Dice loss is particularly effective in addressing class imbalance in medical image segmentation.

Other commonly used loss functions in segmentation tasks include the *Jaccard Loss* that measures the intersection over union of the predicted segmentation and the ground truth, the *Perceptual Loss* that computes the difference between high-level features of images rather than differences between pixels and the *Total Variation Loss* that penalizes differences between adjacent pixels, encouraging spatial smoothness in images.

3.6.5 Optimization algorithms

Optimizers are another basic concept of DL training process. They are algorithms that iteratively adjust network parameters to minimize the loss function by using gradients and improve the learning process. These gradients show how parameters should be adjusted, and the update process is repeated until the loss converges or a maximum number of iterations is reached.Among the most widely used are Stochastic Gradient Descent (SGD) and Adam, each offering specific advantages strengths depending on the context of the task.

SGD is the foundational optimizer, updating parameters by estimating gradients on small, randomly-selected subset of the data. Despite its simplicity, SGD remains highly effective, particularly when paired with momentum and weight decay. Momentum is a term added to the update rule, that helps the optimizer to continue moving in the same direction even if the local gradient is small. Apart from its computational efficiency, it provides strong generalizations abilities for models specialized in vision tasks [68].

On the other hand, Adam (Adaptive Moment Estimation) builds upon the RMSProp optimizer and momentum by maintaining separate adaptive learning rates for each parameter, based on moving averages of past gradients and their squares. This makes Adam fast, robust to noisy gradients, and typically requires less hyperparameter tuning.

Choosing between them often comes down to a trade-off, as Adam converges faster, while SGD with momentum often results to better long-term performance and generalization [69].

Another critical component is learning rate scheduling, which refers to the dynamic adjustment of the learning rate during training. By modifying the model's parameters in response to the error determined for the training data, the learning rate hyperparameter regulates how quickly a model learns. It can greatly affect convergence and stability. The optimizer can balance exploration and refinement by beginning with big updates and finishing with fine-grained modifications when learning rate scheduling is implemented appropriately.

Chapter 4

Segmentation Tasks in Medical Imaging

4.1 Medical imaging

4.1.1 Introduction

The fundamental Deep Learning components and techniques discussed in Chapter 3, can slightly vary depending on the DL task. In this chapter we are discussing the specific components and methods that are connected with image segmentation tasks in medical imaging in order to better understand the specific needs and requirements of the methodology adopted in this project.

4.1.2 Fundamental concepts

Imaging is a fundamental component of modern medicine with a crucial role in screening, diagnosis, treatment and surveillance. It refers to the visualization of internal body structures through various modalities like CT, MRI and PET. Almost every patient has undergone at least one type of radiological examination at some point. Radiology advancements have made significant changes in healthcare, from a thorough ultrasounds of a fetus to a detailed brain computed tomography (CT) scan to identify a target for therapy after a stroke [70].

[71] highlights that medical image segmentation is a crucial post-processing task in medical imaging, that can strongly affect diagnosis, treatment planning, and analysis of findings. It describes the process of dividing an image into distinct regions that represent anatomical structures or pathological areas, such as organs, tissues, or lesions by differentiating between the foreground and background. Pre-operative planning, organ border delineation, tumor localization, and other crucial tasks are supported by this segmentation process, which enables accurate analysis of medical data. Because of its importance, segmentation accuracy directly affects clinical results, making it a crucial component of contemporary healthcare systems' workflow .

Radiomics is another relevant emerging field in medical imaging that transforms scans into quantitative data through automatic extraction of features. For example in oncology, radiomic analysis of tumors can capture important patterns for heterogeneity, such as variations in cell density, necrosis, fibrosis, or hemorrhage. When combined with image segmentation, these quantitative features allow precise localization and characterization

of tumor regions, with a potential to improve treatment and prognosis in cancer patients [11, 72].

4.1.3 Traditional vs DL Methods

In previous decades, some of the most popular techniques used for medical image segmentation were thresholding, edge-based techniques, region-based approaches, clustering-based methods and graph-based segmentation. These methods are mostly reliant on specified rules and intensity-based operations, making them relatively efficient and interpretable. However, their performance is limited when dealing with complex medical images containing noise, intensity variations, or unclear boundaries. These limitations highlighted the need for more concrete approaches for segmentation tasks.

In the past years, DL methods have surfaced as crucial applications in this field, because of their unique performance in automatic feature extraction and complex data handling. This methods utilize neural network architectures, such as CNNs, FCNs, U-Net and RNNs to automatically identify and delineate objects or regions of interest within images. The main functionality of these models involves optimizing the parameters to accurately map input data to corresponding segmentation masks [71].

As mentioned by [73], DNNs have become an integral part of computer-aided diagnostic (CAD) systems. They offer solutions with identifying patterns and features that cannot be easily observed by radiologists, thus supplying additional information for diagnostic procedures.

Segmentation tasks can therefore be categorized as manual, semi-automated, and fully automated. **Manual segmentation**, requires radiologists or relevant medical experts to delineate anatomical structures by hand, providing quite accurate but extremely time consuming results. **Semi-automated segmentation** utilizes computational tools, like DL applications, to improve efficiency, followed by manual inspection and editing by experts. However, the optimal method is **fully automated segmentation**, based on well-trained DL models FCNs and U-Net, which enable fast and consistent segmentation with minimal expert help required, especially for large and complex datasets [74].

Automated segmentation processes do come with some doubts. First, there is no universally consistent "ground truth" for tumor boundaries, since poor image contrast and adhesion with nearby tissue can lead to subjective interpretations and inconsistencies that can undermine the reliability of machine learning models [75]. Additionally, segmentations might still vary across different time points, physicians and ML algorithms. These differences can strongly affect diagnostic conclusions and treatment planning. Therefore, consistency in segmentation results, or in other words reproducability, remains a great concern and the ultimate goal of research on these methods [40].

4.2 Semantic vs instance Segmentation

Semantic segmentation is a foundational technique in computer vision tasks that focuses on classifying each pixel (or voxel) in an image into specific categories or classes, such as objects, parts of objects, or background regions. Given a new image, the algorithm should output the pixels of the image that belong together semantically. This method provides a universal understanding of the image by breaking it into meaningful distinct regions based on the content and context of the scene [76].

The workflow of semantic segmentation process begins with the analysis of labeled training data for better understanding of object classes and patterns. Afterwards, a semantic segmentation network with convolutional layers for feature extraction and upsampling layers for dense classification is implemented. The network is then trained to capture pixel-wise classification and optimize segmentation accuracy using loss functions. Lastly, the trained model processes unseen images and generates segmentation masks by classifying each pixel into specific semantic categories.

Other segmentation techniques include instance segmentation, which offers a more detailed understanding of the image by differentiating between individual objects. It gives each object instance a unique label, and disregards individual classes, in contrast to semantic segmentation, which divides each pixel into broad categories without differentiating between instances of the same class [77].

4.3 Imaging modalities

4.3.1 Main categories

In the contemporary diagnostic landscape, the most widely used imaging modalities from clinics and hospitals based on [78], are the following:

- **X-ray:** X-ray imaging is one of the oldest and most widely used medical imaging modalities. Ionizing radiation is used to capture internal body structures, especially thick tissues and bones. X-rays are quick, affordable, and helpful in the diagnosis of lung disorders, infections, and fractures. However, there is a limit to the vision of soft tissues, and repeated use of radiation offers a health risk.

- **Computed Tomography (CT):** By merging several cross-sectional images, CT scanning evolves X-ray technology to provide detailed 3D reconstructions of internal regions. Since, it offers improved depiction of soft tissues, blood arteries, and bones, it is ideal for cases involving cardiovascular issues, cancer, trauma, and stroke. Compared to traditional methods, its capacity to identify minor variations in tissue density makes it an invaluable diagnostic tool. Despite its advantages, CT is worse at soft tissue discrimination than MRI, while exposing patients to higher radiation doses. However, thanks to recent developments like low-dose CT and contrast-enhanced scans, which have increased safety and diagnostic potential, CT remains a fundamental component of contemporary medical imaging.

- **Magnetic Resonance Imaging (MRI):** MRI uses strong magnetic fields and radio waves to produce high-resolution images of soft tissues. It is irreplaceable for the imaging of the brain, spinal cord, and musculoskeletal system. Unlike X-ray and CT, MRI is safe for multiple sessions per patient than X-ray and CT, as it does not involve ionizing radiation. However, it is expensive, time-consuming, and ineffective for imaging structures containing air or bone.

- **Ultrasound:** Ultrasound imaging relies on high-frequency sound waves to produce dynamic, real-time visualization of organs and blood flow. It is widely used in obstetrics, cardiology, and abdominal imaging, due to its mobility, cost-effectiveness and safety. The main disadvantage is increased human error probabilities by the operator and limited accuracy for bones or gas-filled structures.

- **Nuclear Imaging:** Nuclear imaging, including PET and SPECT, involves the injection of radiotracers to visualize physiological and metabolic processes. It is utilized to detect functional abnormalities before structural changes occur, in the fields of oncology, neurology, and cardiology.
- **Electrical Impedance Tomography (EIT):** EIT is a non-invasive imaging technique that uses surface electrodes to reconstruct conductivity distributions within the body. It is safe, portable, and radiation-free, with promising applications in lung function monitoring, breast cancer detection, and brain imaging.

Recent advances in the imaging field include contrast-enhanced MRI and cardiovascular imaging, methods that aim to improve specificity, resolution, and functional insights, addressing limitations of traditional modalities. AI and data mining techniques are also being incorporated in automated image analysis, to enhance diagnostic precision and efficiency.

4.4 Annotation standards

As described in Chapter 3, supervised learning relies on the availability of accurately labeled training data. The development of a supervised learning algorithm requires a function able to map each training data point to the corresponding label. Self-supervised learning techniques also utilize labeled data typically as a second step after training a model on automatically generated pseudolabels [45]. In medical imaging, this mapping between data and labels is implemented through **annotations**, which is the process of highlighting specific features or structures within the images, through bounding boxes, segmentation masks, or labels to define the ground truth to be used for training models.

The main annotation forms for image processing tasks are:

- **Categorical labels**, where a single class is assigned to an entire image, volume, or patient.
- **Segmentation masks**, where the image is divided in regions corresponding to pathological regions or anatomical features.
- **Regions of interest (ROIs)**, providing the location and size of specific structures, often accompanied by class labels.
- **Landmark coordinates**, indicating precise anatomical points of reference.

While the standard image processing packages such as ImageJ and 3D Slicer are commonly used to generate image annotations, dedicated software allowing for a fast workflow has been developed and made available, both by the free software community and for commercial purposes [79].

Image annotations are usually created through image processing applications and software. 3D Slicer is one of the most common programs used for complex volumetric data, as it provides a user-friendly, yet complete environment for 3D visualization, segmentation, and quantitative analysis. Another flexible, open-source tool for image processing, visualization, and manual labeling is ImageJ, which is designed for 2D and basic 3D images. For semi-automatic segmentation tasks, ITK-SNAP is an excellent option, providing region-growing and manual contouring tools, crucial for correctly labeling medical images.

4.5 Segmentation dimensionality

Deep learning methods for medical image auto-segmentation have been developed to process various input image types, like 2D, 2.5D, or 3D formats. These different approaches were created to handle different output formats from CTs and MRIs, according to the requested task or region of interest. The differences between the implementation and strategies for each input dimensionality type, as discussed by [80], are presented below.

For the 2D implementation, one slice per medical image is processed at a time. All feature maps and parameter tensors in the model are also 2D, and the output is a segmented 2D slice. This method requires the least memory, making it computationally efficient, but the lack of spatial context between neighboring slices, can lead to critical information being missed.

The 2.5D approach was developed as a middle ground solution between 2D and 3D. It analyzes five consecutive slices of the image simultaneously, using the center slice as the target for segmentation while utilizing information from the adjacent slices.

In 3D segmentation, the entire image volume or selected crops of the volume are processed in 3D space. The model is able to apprehend all spatial context throughout the entire volume data, as feature maps and tensors are in 3D. The volumetric segmentation mask produced is also 3D and each voxel corresponds spatially to a voxel in the input, allowing for accurate detection of structures across height width and depth dimensions. Thus, this method works best for complex segmentation tasks like brain and tumor segmentation or localization. However, it requires up to 20 times more memory than 2D or 2.5D models, making it much more expensive and time-consuming.

4.6 Segmentation process

As already discussed, image segmentation with specific CNNs is a key DL application that has improved accuracy and automation in the fields of diagnosis, treatment and medical data analysis. These models can segment and detect specific anatomical structures, after being trained with large and correctly labeled datasets from modalities like CT and MRI [73].

4.6.1 Overview of popular segmentation tasks

Some of the most widely known applications of DL segmentation in medical imaging are the following:

- Organ segmentation: isolation of different organs (e.g., liver, heart, or brain) to assist in surgery or therapy planning. Example: [81].
- Tumor and Lesion Segmentation: identification of tumors or lesions within organs for quantitative analysis, prognosis, and evaluation of treatment plans. Models have been developed for brain, liver and beast cancer or cancers that have metastasized in different regions. Example [61, 82].
- Vessel Segmentation: extraction of vascular structures for blood flow analysis, abnormality detection, or therapy guidance.

- Cell and Histopathology Segmentation: applied to microscopic or histopathology images to separate cells or tissue structures for computational pathology. Example: [83].
- Multi-Organ Segmentation: Localization of multiple organs at the same time for complete anatomical mapping, especially helpful in oncology or radiotherapy procedures. Example: [8].

4.6.2 Key concepts in the segmentation procedure

Medical image segmentation techniques involve some specialized concepts that need to be handled carefully, in addition to the fundamental DL techniques.

To better understand how segmentation models operate, several concepts should be introduced prior to implementation details:

Feature learning is the process by which CNNs automatically extract complex features from medical images. While deeper layers capture high-level semantic information, like organ boundaries and position or lesion shapes and intensity, low-level layers pick up fundamental characteristics like edges or textures.

CNNs are frequently trained for classification and *anomaly detection* tasks in addition to segmentation. This refers to the finding of abnormal patterns that differ from typical anatomy or expected physiological signals in the medical data captured from CTs and MRIs. It is an important requirement for the development of complete and accurate datasets for automated segmentation tasks.

As mentioned already, the *segmentation process* results to the comparison of the ground truth (GT) mask, annotated by experts, which serves as the reference point and predicted masks by the trained model. This comparison takes place during the evaluation phase using metrics such as Dice Similarity Coefficient (DSC) or Intersection-over-Union (IoU) that showcase the performance and possible limitations of the model in segmenting the target regions.

Another important aspect for segmentation performance are scale and intensity adjustments. As explained by [84], contrast enhancement, scale intensity changes and random intensity augmentations are applied during preprocessing to make the requested anatomical structures, such as small tumors with slight intensity variations from their environment, much more clear and visible for the model.

For CT images, intensity values that correspond to tissue density are measured in Hounsfield Units (HU). The process of adjusting those values to better depict the regions of interest for a specific task is called HU Windowing. This process is implemented by selecting the appropriate range of HU values to highlight relevant regions and suppress irrelevant background noise, improving both training and inference [85]. Figure 4.1 shows the HU values for various structures and elements captured by CT scans, along with the resulting images after windowing. For example, for liver-specific segmentation tasks, a window ranging approximately from -150 to 400 is suggested by numerous studies [85,86].

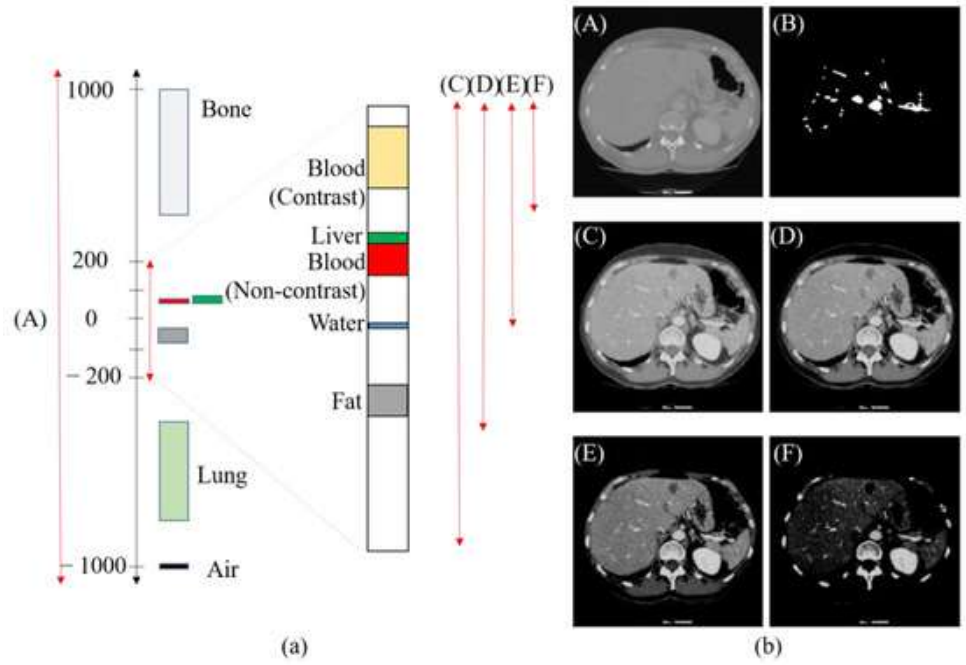

Figure 4.1: Distribution of HU values for key anatomical structures and components [5]

The last method introduced is cropping, which involves focusing the model on the most important area of the scan by isolating it from the background. This approach is quite useful for tasks where ROIs occupy a small part of the overall image volume, like tumor segmentation [87]. It also helps reduce the input size and the overall computational load, making the training process more efficient.

4.7 Challenges and considerations

4.7.1 Class imbalance

One of the most common problems in medical image segmentation with DL is class imbalance. This refers to datasets where the structure to be segmented takes up a significantly smaller number of voxels than the background. Thus, the network is fed with a great percentage of irrelevant information during training, resulting in suboptimal performance, as suggested by [88]. More specifically, the network overfits to the few and under-represented foreground samples found in the data, decreasing its generalization ability. As shown by [89], in training experiments with class imbalance, the logit activations, hence the raw outputs before applying activation functions during testing, tend to move across the decision boundary, causing the model to miss smaller structures. At the same time, predictions for the well-represented classes remain stable. This problem is quite critical in tasks like tumor segmentation, since tumors are usually small and can vary in size, shape, and location, making precise predictions much more challenging.

To minimize the effect of this phenomenon several strategies have been tried. [88] suggested the implementation of specialized loss functions that shift the model's focus towards the minority class during optimization. These loss functions include Generalized Dice Loss, Focal Loss, the (Focal) Tversky Loss, and the Unified Focal Loss.

Another frequent technique is the implementation of sampling strategies. Oversampling is the process of randomly replicating samples from the underrepresented class to

increase their presence. It increases the model's exposure to rare cases, however it reduces the variance of the dataset, since it adds duplicate inputs, increasing the risks of overfitting. On the other hand, undersampling refers to the reduction of the majority class through random downsampling. This can also increase the risk of a biased model, since it excludes unique information from the majority class.

Adjusting class weights before, or even during training is another method used to increase the influence of the minority class in the loss calculation. Specifically, false negative results in the minority class are heavily penalized in an effort to improve the recall metric of the model.

Dealing with class imbalance issues can strongly improve the final performance and results of the segmentation model. It is a problem that was quite important for this project and the methods implemented to tackle it are discussed in later chapters.

4.7.2 Other challenges

The annotation process of medical imaging data also presents several challenges. Privacy and security are major concerns, as medical data is highly sensitive and subject to strict regulations. In addition, specialized knowledge of radiologists or healthcare professionals is often required for the accurate annotation of complex medical data to avoid inconsistencies and misclassifications. Lastly, specialized formats, such as DICOM (Digital Imaging and Communications in Medicine), are required to store and use the data effectively.

Chapter 5

Review of Applied Architectures and Strategies in Liver and Liver Tumor Segmentation

After having discussed the main theoretical concepts of deep learning, medical imaging and segmentation tasks, it is time to take a closer look at the strategies and methods implemented for liver and liver tumor segmentation tasks, which constitute the main topic of this project. For this purpose, this chapter presents the most commonly used network architectures for liver-specific tasks, some of the best datasets used and a comparison of numerous state-of-the-art architectures on datasets for liver and liver tumor segmentation. Of course, the main focus is given on studies regarding CRLM and CT scans, however similar approaches for liver cancer or different modalities like MRI are also included to provide a broader picture.

5.1 Key network architectures

5.1.1 U-Net and variations

Original U-Net

As already mentioned the U-Net is the fundamental CNN-based architecture for medical image segmentation, and was first presented by [6]. It is based on an encoder-decoder structure and takes its name from its shape as shown in Figure 5.1. The encoder is used to extract high-level features from the input image, while the decoder is used to upsample intermediate features and produce the final output. They have an identical structure with the opposite functionality.

More specifically, each level of the encoder consists of repeated 3×3 convolutional layers with ReLU activation for feature extraction, followed by a max pool layer that downsamples the image for the next level. After each downsampling the channels are doubled to make up for the loss of spatial dimensions. level features. Similarly, the decoder restores the spatial resolution of the features that was lost during the encoding, by using similar convolutional layers with ReLU activation, followed by the upsampling operation with a 2×2 transposed convolution layer (up-conv) that reduces the channels in half.

However the most important property of this architecture are the 'skip connections'

that concatenate the feature maps from the encoder to the corresponding layer of the decoder at the same resolution, so that the convolutions are applied to both. This happens because decoder features include deeper semantic information about the characteristics of a specific region of the image, while encoder features provide crucial spatial information from shallow layers, on the exact pixel-wise location of each characteristic. Lastly, the bottleneck is the part where the encoder and the decoder meet and the intermediate features are passed from one to the other through convolutions.

With this simple but functional structure this architecture enables both contextual understanding and spatial precision, which are critical in medical image segmentation tasks [60].

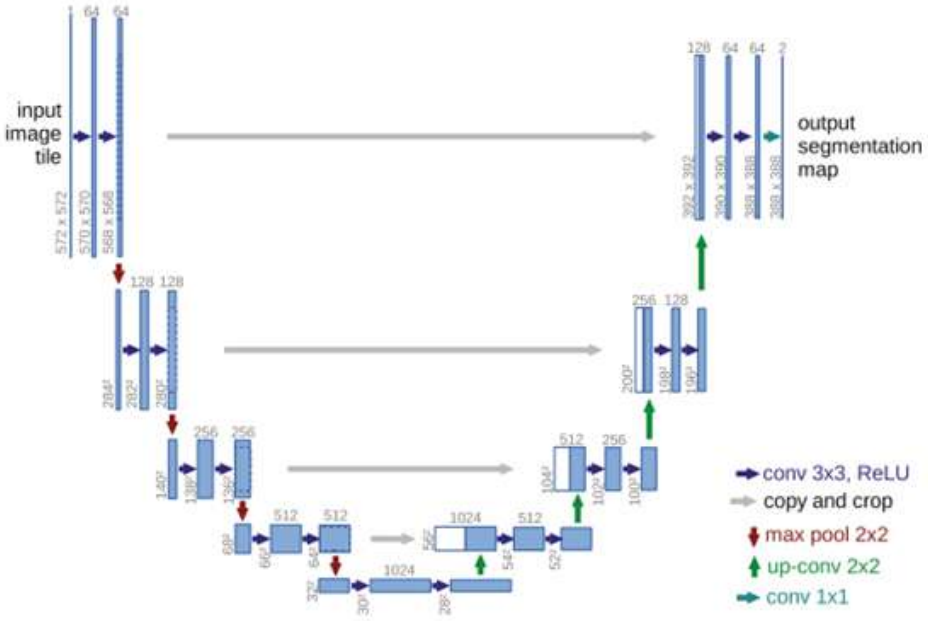

Figure 5.1: The structure and specific layers of the U-Net [6]

The specific 3D implementation of the U-Net was proposed by [90]. All operations (convolutions, pooling and up-convolutions) are extended in the third dimension. $3 \times 3 \times 3$ convolutions are used during compression along with $2 \times 2 \times 2$ max pooling and $2 \times 2 \times 2$ convolutions during upsampling. That way, context on the depth axis is captured resulting in more effective segmentation of complex volumetric structures like tumors. However, for this computation, additional GPU memory and processing times are required.

5.1.2 Residual models

Residual U-Net

[91, 92] proposed a variation of U-Net that incorporates residual blocks into the original architecture. Residual blocks utilize another form of shorter skip connections, which add the activation of a layer to the output of further layers, disregarding intermediate transformations. This technique allows the model to learn identify mappings when needed and helps tackle the issue of vanishing gradients. Residual Networks are created by multiple stacked residual blocks.

Based on this method, the ResUNet proposed replaces the standard convolutional layers of the original approach with residual blocks. Both long skip connections between the encoder and the decoder and shorter residual skip connections exist in this architecture. The result combines the training efficiency and stability of residual connections with U-Net's feature extraction abilities, increasing the training performance on deeper networks.

SegResNet

Another successful architecture was proposed by [7]. The SegResNet combines the original encoder-decoder approach, with an autoencoder architecture. Figure 5.2, analytically shows the architecture implemented. It is evident that a bigger encoder is used for enhanced feature extraction, while each green block refers to a residual block. Apart from the decoder which outputs the segmentation of the input with the same spatial size, the other branch of the encoder leads to a Variational Autoencoder (VAE) that reconstructs the input image only during training and is used to enhance the encoder's regularization. For context, VAEs are are generative models that create data similar to the input used during training. They learn a continuous probabilistic representation of the low-level features of the data they compress and reconstruct.

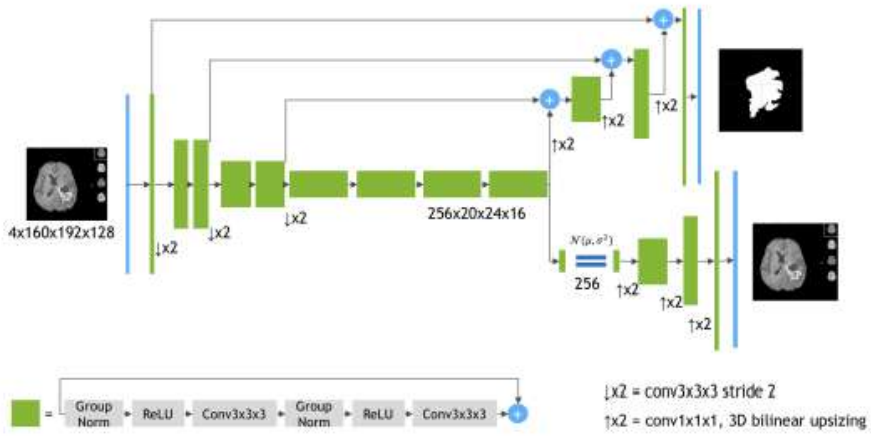

Figure 5.2: SegResNet architecture with VAEs [7]

5.1.3 Transformer models

Transformer architectures originally became popular in the field of natural language processing, however their unique ability to capture complex global relationships within data has made them quite useful in computer vision tasks. In contrast to CNNs, Visual Transformers (ViT) utilize the attention mechanism to process the entire input and learn relationships between distant regions of an image. More specifically, multi-head self-attention enables the model to focus on multiple spatial regions simultaneously, allowing it to capture diverse contextual interactions at the same time. This ability tackles the spatial limitations of CNN models, that focus on local feature extraction [93]. These models have demonstrated excellent performance on computer vision tasks, while relying on extensive pretraining with large datasets, which enables them to generalize well after fine-tuning. Another unique characteristic of ViTs is their token-based processing,

as images are divided into patches, linearly embedded, and then treated as sequences of tokens.

UNETR

[8] approached the task of 3D medical image segmentation, as a "sequence-to-sequence prediction problem" and created the UNEt TRansformers (UNETR) architecture. The original U-Net shape is used, with a transformer serving as the encoder to learn sequence representations of the input volume and capture the global information. The attention heads within the transformer encoder learn complex dependencies across distant regions of the 3D volume, which is particularly important for medical imaging where lesion locations can vary significantly. Skip connections are also used to connect the transformer encoder with the decoder and facilitate the computation of the final output.

The shape and layers utilized are shown in Figure 5.3.

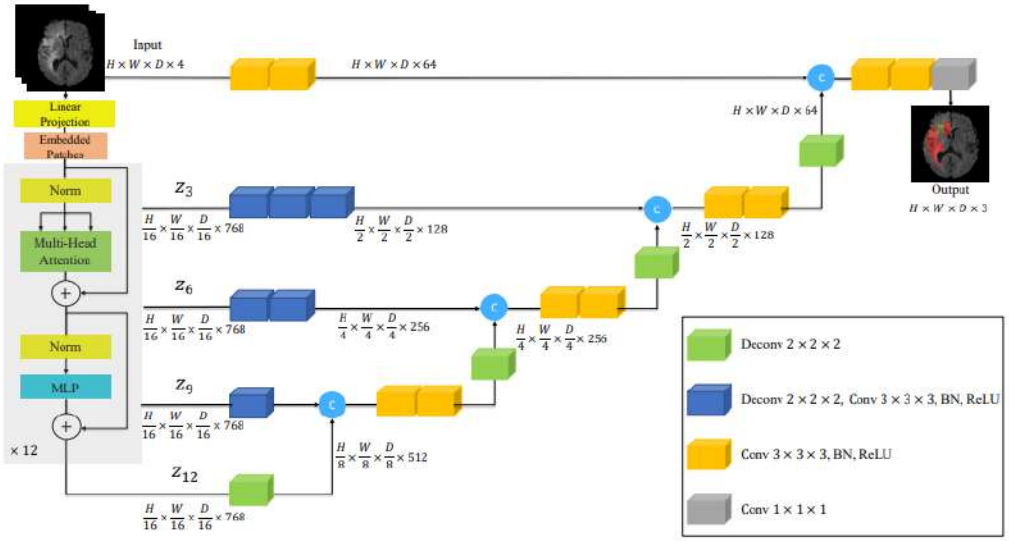

Figure 5.3: UNETR Architecture [8]

SwinUNETR

A more complex transformer-based implementation was proposed by [9] for brain tumor segmentation. Unlike the UNETR approach, here a **S**hifted **win**dows or SWIN transformer is used at the encoder for feature extraction at five different resolutions. The Swin method divides the image into discrete local windows and allows cross-window connection. Self-attention is then computed only within each smaller window and not globally for all patches of the picture like the ViT approach. **By applying self-attention in a hierarchical and localized way, the model effectively reduces computational cost while preserving the ability of attention heads to model relevant dependencies. Information is exchanged by adjacent windows, enabling the model to capture local and global context progressively, while keeping the computational complexity reduced. The encoded feature representations in the Swin transformer are fed to A CNN-decoder receives the feature representations from the encoder via skip connections at multiple resolutions and then reshapes them and passes them through a residual block. The upsampling continues until the final segmentation output. The complete SwinUNETR structure is presented in Figure 5.4.

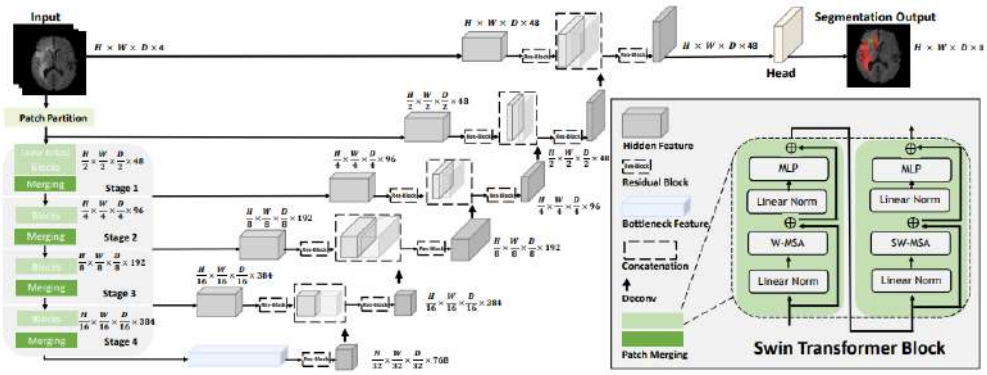

Figure 5.4: SwinUNETR Architecture [9]

5.1.4 Other network variants

Attention U-Net

A different idea, still based on the original U-Net structure, was proposed by [10]. The architecture, shown in Figure 5.5, uses a similar encoder to U-Net, while adding an attention gate (AG) at the end of each skip connection from the encoder to the decoder.

For context, attention mechanisms are divided in hard attention, which crops image to highlight the region of interest, and soft attention, which implements weighting to showcase the relevance of the different regions of the image, making it trainable with backpropagation.

The AGs used at each skip connection apply soft attention to reduce activations for irrelevant regions and block irrelevant low-level features from being passed to the decoder. They use contextual information from lower-resolution layers to guide the network towards important areas of the image, enabling the model to better recognize structures of different sizes and shapes, like tumors, during training.

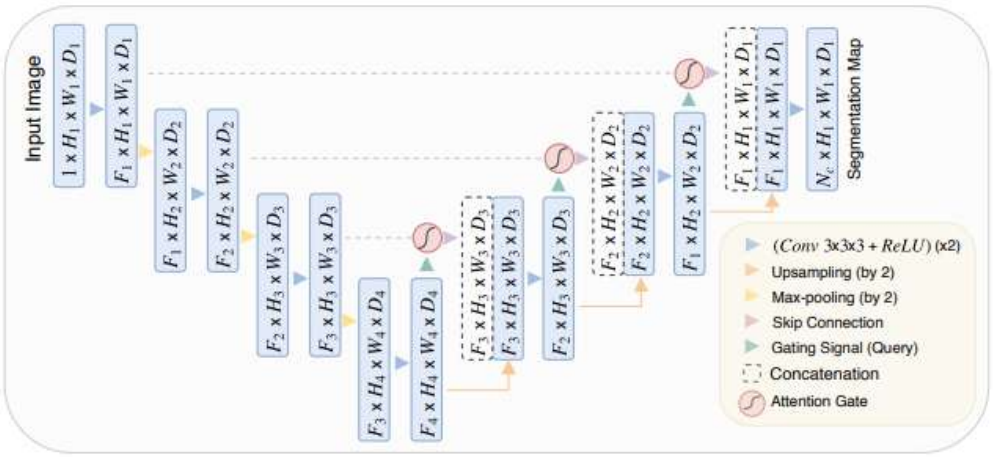

Figure 5.5: Attention U-Net Architecture [10]

Cascaded FCN

Another structure proposed by [94] were cascaded FCNs, hence a series of stacked FCNs. In this approach, each model takes advantage of the contextual features extracted by

the previous model's prediction map. For the task of liver segmentation, two FCNs were cascaded, where the first focused on liver as the ROI to be used by the second FCN which performed liver lesion segmentation.

V-Net

[95] proposed the V-Net architecture, a 3D FCN design for volumetric medical image segmentation, based on the U-Net structure. The main difference from the U-Net is the use of strided convolutions for downsampling, instead of traditional pooling. These convolutions facilitate feature extraction along with resolution reduction with lower memory usage, as no switches mapping the output of pooling layers back to their inputs are needed for back-propagation. Additionally, at each stage of the compression path, residual functions are implemented, ensuring convergence. In the expansion path, deconvolutions are used to recreate the initial resolution, resulting in a two-channel volumetric segmentation map.

U-Net ++

U-Net++ is an advanced segmentation architecture built on the original U-Net design. Additional skip pathways between the encoder and decoder paths are introduced for an enhanced connection between the two. These pathways consist of convolution layers for better semantic connection between the encoder and decoder, as well as dense skip connections that improve gradient flow. Additionally, deep supervision is introduced, which refers to the calculation of loss at intermediate layers of the network on top of the final layer. These extra loss connections help earlier layers to learn more discriminative features, improving model accuracy and enabling faster convergence. With these advances, U-Net++ captures descriptive details of target structures more effectively.

5.2 Comparative analysis of strategies and datasets from similar work

At this point, the methods and results of several similar papers are going to be presented. This review is necessary to evaluate the usual practices and expected results for tasks similar to this project. Valuable strategic insights were taken from these papers. The specific strategies and limitations that were observed are presented below, followed by a comparison of the results and methods used in Table 5.1.

The datasets that were used in the papers selected for this comparative analysis are the following:

- The Beyond the Cranial Vault (BTCV) dataset [96] contains 30 contrast-enhanced abdominal CT scans collected in the portal venous phase.
- The Medical Segmentation Decathlon (MSD) [97] was a biomedical image analysis challenge which included different tasks with data from multiple anatomical regions and modalities, such as brain, heart, hippocampus, liver, lung, pancreas, prostate, colon, spleen and hepatic vessels.
- A similar approach focused on the liver is the Liver Tumor Segmentation (LiTS) dataset by [61], which includes contrast-enhanced abdominal CT scans. The data

varies in resolution, contrast and slice thickness, due to the numerous institutions and scanner types used for the collection of the data. The 3DIRCADb Dataset, is a subset of 20 patients from the LiST dataset, with 75% of patients including hepatic tumors.

- The CHAOS (Combined Healthy Abdominal Organ Segmentation) dataset, created by [98] contains abdominal MRI and CT images from 80 patients. The dataset mainly comprises of healthy abdominal organs, making it ideal for pretraining or baseline organ segmentation tasks.
- The CAIRO5 dataset originates from the CAIRO5 phase 3 clinical trial, conducted by the Dutch Colorectal Cancer Group [99]. It consists of 407 patients with unresectable CRLM lesions, who were treated with systemic therapies based on tumor genetics.

Numerous papers with relevant segmentation tasks were found. The discarding criteria were the use of different modalities (MRI or histopathological images), the focus on different types of cancer, the implementation of too specific architectures or training strategies and finally the lack of full access to the entire article. All selected papers used CT scans mainly during training and focused on liver and liver cancer segmentation and employed the U-Net architecture or variations of it to train the models. For example, [100] proposed the "Hybrid W-Net" structure, which takes advantage of 2D features from a pretrained DenseNet121, which reuses previous convolutions, leading to a lower number of filters. These extracted features are then passed on a 3D DenseNet and then the entire architecture is trained from the beginning.

Most of the papers pinpointed the challenges of this task and the under-representation of the tumor class. Hence, preprocessing and augmentative techniques were used to increase tumor samples. Moreover, many studies [85, 101, 102] excluded slices without liver or tumor instances for reduced memory usage and better performance results.

Other strategies proposed the exclusion of poor quality patient volumes [103] or the use of data sets from patients with previously unresectable lesions or generally large tumor samples who then underwent treatment [104, 105]. pretraining and transfer-learning techniques were also implemented especially for CRLM cases. Some of the models were pretrained on large public datasets [51, 105], like LiTS, while others created automated pipelines that passed on segmentation masks created from the liver segmentation model, to the tumor segmentation model [104, 106].

Some papers implemented internal datasets from medical trials in their training process. Expert radiologists were assigned to edit and approve these datasets, which were used to enhance the testing of the model's performance [101] or increase the model's specialization on CRLM lesions [104, 105].

Another important limitation was the admission that the increased dice values observed derived from the great class imbalance between tumors and background, meaning that some studies included background in the dice calculation, which was easily segmented and therefore inflated the metric results [51, 85, 91]. Moreover, both global and per-case dice scores were reported. Generally, for clinical applications, per-case dice is more important, as it evaluates performance distinctly for each patient, instead of global dice which considers all predictions and labels as one big volume.

Regarding CRLM segmentation tasks, research showed that performance drops significantly for smaller lesions [103], which are quite common in these datasets, and that is why dice was presented according to lesion size [100, 105].

The hardware specifications ranged from different GPU variations, including NVIDIA T4 x2 GPU, the NVIDIA-Tesla V100 32GB GPU, the Tesla V100 GPU, and the 8 GB NVIDIA GeForce RTX 2080. These are all powerful GPUs, necessary for heavy models and 3D data.

Table 5.1: Comparative summary of liver and liver tumor segmentation studies.

Author (Year)	**Model**	**Dataset**	**Liver Metrics**	**Tumor Metrics**	**Notes**
Muhammad et al. 2024 [85]	ResUNet	MSD Task03-Liver	DSC: 0.9837 Acc: 0.98	DSC: 0.871 Accuracy: 0.95 Precision: 0.93 VOE: 12.09	Background included for DSC | Slices excluded | Separate liver & tumor segmentation
Özcan et al. (2023) [101]	AIM-UNet (UNet with inception module hybrid)	CHAOS LiST Internal Data	DSC: 0.979 (CHAOS)	DSC:0.756 (LiST) DSC: 0.655 (3DIRCADB)	8 GB NVIDIA GeForce RTX 2080 | liver-free slices excluded | 6 different views created from each sample
Rahman et al. (2022) [91]	ResUNet	3DIRCADb	Acc: 0.9923	Acc: 0.9927	All tumor masks combined | Class imbalance and overfitting increased accuracy for tumors
Yashaswini et al. (2025) [102]	ResUNet	3DIRCADb	DSC: 0.9144 Acc: 98.18 Preciison: 84.42	DSC: 0.76 Acc: 0.98 Prec: 0.62	Images lacking liver and tumor were removed
Manjunath et al. (2022) [107]	Modified UNet (58 layers)	3DIRCADb	DSC(LiTS): 0.9615 DSC(3Dircadb): 0.9194	DSC(LiTS): 0.8938 DSC(3Dircadb): 0.698	Testing from open data excluding images without ROI
Arora et al. (2025) [51]	U-Net ResUNet Attention UNet Cascade UNet	3DIRCADb	—	DSC(Casc UNet): 0.92 DSC(AttUNet): 0.91	NVIDIA T4 x2 GPU | potentially background included | models pretrained on public tumor datasets

Continued on next page

Table 5.1 – *Continued from previous page*

Author (Year)	**Model**	**Dataset**	**Liver Metrics**	**Tumor Metrics**	**Notes**
Wesdorp et al. (2023) [104]	U-Net	CAIRO5	DSC(glob and median): 0.96	DSC(glob): 0.86 DSC(median): 0.8 DSC(val):0.6 Prec:0.89 Recall:0.84	CRLM focus | Auto pipeline| patients with a increased metastasis
Bereska et al. (2024) [106]	nnU-Net	CAIRO5	—	DSC(Internal): 0.85 DSC(External): 0.83	Self-learning teacher-student framework | COlorectal CAncer Liver metastasis Assessment (COALA) model
Anderson et al. (2022) [100]	U-Net ResUNet DenseUNet Hybrid WNet	3D-IRCADb01	–	DSC: (<15mm): 0.16 (>15mm): 0.74 MSD: (<15mm): 28.3 (>15mm): 1.23 Sensitivity: (<15mm): 0.23 (>15mm): 0.98	training distributed into "slabs" | 7% mean sensitivity in sites <10 mm | NVIDIA-Tesla V100 32GB GPUs

Continued on next page

Table 5.1 – *Continued from previous page*

Author (Year)	**Model**	**Dataset**	**Liver Metrics**	**Tumor Metrics**	**Notes**
He et al. (2021) [105]	Residual Attention U-Net	LiTS local dataset	DSC: 0.95	DSC(Portal venous phase): 0.73 Sensitivity: 0.82 Precision 0.44	Local dataset with patients after RFA or MWA treatment | 2D U-Net fro liver & 3D U-Net for tumors | Models pre-trained on LiTS | Equal number of foreground and background patches | Tesla V100 GPU's | Enhanced precision and F1 for lesions greater than $0.5cm^3$

Part II

Experimental Work and Methodology

Chapter 6

Methodology and System Architecture

6.1 Overview of the proposed approach

The main purpose of the experimental part of this project was to create a robust and accurate model for liver and CRLM 3D segmentation from CT scans. During this process many pre-built implementations from the MONAI framework were leveraged for simplifying development and accelerating implementation. Many experiments were conducted, regarding different dataset handling methods, like cropping, network architectures, preprocessing techniques, evaluation metrics and hyperparameter tuning to finally come up with the ideal combination for this specific task.

Most of the trials were implemented on top of a local NVIDIA GeForce GTX 1650 Ti 4GB GPU. However, since its capacity limited performance in some cases, some experiments were conducted remotely with an NVIDIA GeForce RTX 4080 16GB GPU. The various methods and techniques that were developed and tested for this project are thoroughly presented in this chapter.

6.2 Tools and frameworks used

A combination of specialized DL frameworks and supporting tools were used to facilitate the coding implementation, result visualization and variety of experiments conducted in this project. The key frameworks include MONAI, PyTorch, Google Colab, and Weights & Biases (W&B).

6.2.1 MONAI

The Medical Open Network for AI (MONAI) is an open-source framework, built on top of PyTorch, specifically designed for DL in medical imaging. It provides a complete set of tools for the entire pipeline of medical imaging tasks, including preprocessing, augmentations, model architectures, loss functions, and evaluation metrics. The ready-to-use network implementations, data transforms and loss functions that are offered by MONAI and are customized for 3D medical images, simplified greatly the implementation of the training pipeline.

Moreover, MONAI ZOO, which offers a variety of pretrained models for medical tasks,

was also utilized for fine-tuning experiments. Overall, MONAI made the experimentation with different architectures, parameters and methods, much quicker and simpler.

6.2.2 PyTorch

PyTorch is a deep learning library built on Python. It provides GPU acceleration, dynamic computation graphs and other important tools for DL developers. It is also the basic library that is used for MONAI implementations. It facilitated seamless integration with CUDA for GPU acceleration, which is crucial for handling heavy 3D volumetric CT data, while also providing more low-level tools for handling neural network layers during fine-tuning.

6.2.3 Google Colab

Google Colab provides a cloud-based and friendly-to-use development environment. Its GPU resources were utilized for some experiments, however they were not enough for full training runs. On the other hand, it provided a useful data visualization tool for inspecting CT slices, verifying preprocessing functionality and generating plots during the evaluation phase of the model.

6.2.4 Weights & Biases (W&B)

W&B is a machine operations platform, primarily used for tracking and visualizing model performance. It was used to log the different experiments, visualize training metrics such as accuracy, loss, and dice and comparing the different methods and approaches implemented. W&B automatically logged key metrics, enabling real-time monitoring of model performance. Moreover, it was also used to log offline runs from a virtual machine that was used for some experiments. Overall, it was especially valuable for comparing and identifying the best training strategies and configurations and presenting the final results.

6.3 Dataset used

The dataset used for the purposes of this project was derived from [11]. This dataset represents the largest compilation of segmented, portal-venous, hepatic CT scans for image analysis of CRLM.

CT scans are used during surgical evaluation to determine the feasibility and process of the procedure to remove the hepatic tumors. This resection must be accomplished with adequate future liver remnant (FLR) for liver regeneration. The dataset includes preoperative hepatic CT scans, clinicopathological data, and recurrence or survival data from 197 patients who underwent hepatic resection of CRLM. For each patient, segmentations of the liver, vessels, tumors, and future liver remnant (FLR) were created.

Patients were selected from 384 consecutive hepatic resections previously utilized for two unrelated studies, based on strict inclusion and exclusion criteria. All patients had pathologically confirmed resected CRLM and available pathologic analysis data of the underlying non-tumoral liver parenchyma and hepatic tumor. Moreover, they all contained preoperative conventional portal venous contrast-enhanced multidetector CT (MDCT) performed within 6 weeks of hepatic resection. Patients with 90-day mortality or less

than 24 months of follow-up were excluded. Also excluded were patients who received preoperative hepatic artery infusion (HAI), underwent local tumor ablation, had more than three wedge resections in the FLR, or had no visible tumor on preoperative imaging.

Each patient had a conventional portal venous phase contrast-enhanced CT scan within 6 weeks after surgery. Segments of the liver, tumors, vessels and bile ducts were generated semi-automatically and used to create a 3D liver model. Resections were virtually drawn on these models using postoperative imaging and pathology. Segmentation was performed by transferring images from the picture archiving and communication system (PACS) to a dedicated workstation, using standard image processing techniques. The liver parenchyma, tumors, vessels, and bile ducts were segmented semi-automatically.

The final segmentations are shown in Figure 6.1. In every CT, the liver is highlighted in green, with the future liver remnant (FLR) distinguished by a darker shade of green, representing the portion of liver expected to remain after the resection. The hepatic (orange) and portal (yellow) veins are segmented to visualize vascular anatomy critical for surgical planning. Each unique metastatic tumor region is presented with a different color (blue, red, purple) and corresponds to a different label file. All above segmentations enable precise modeling and provide a complete analysis of the entire region of interest.

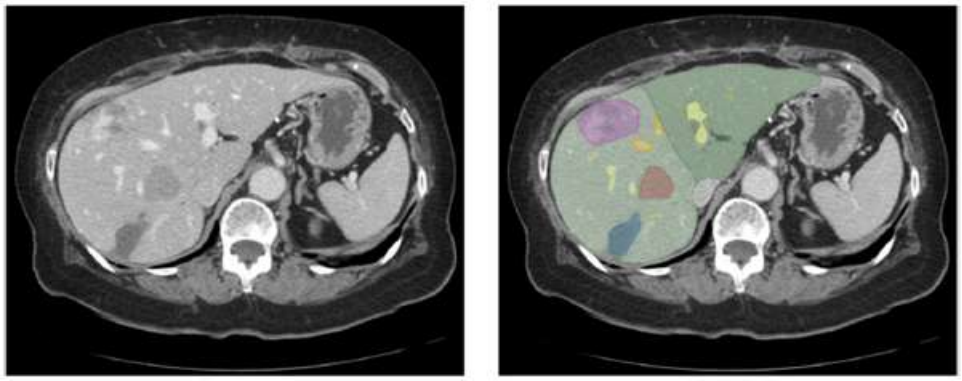

Figure 6.1: Dataset labels [11]

Survival data includes overall survival, disease-free survival, and hepatic disease-free survival. At final follow-up (median 102 months), 90 patients were alive, of which 75 had no hepatic recurrence and 59 had no recurrence of any kind. Median times were 76 months for overall survival, 53 months for hepatic disease-free survival, and 22 months for disease-free survival.

6.4 Data preparation

6.4.1 DICOM to NIfTI transformation

The dataset is publicly available on The Cancer Imaging Archive (TCIA) as "Preoperative CT and Recurrence for Patients Undergoing Resection of Colorectal Liver Metastases." DICOM images and segmentation masks, along with metadata files with clinical, pathology, and survival data, are included in the dataset page.

DICOM (Digital Imaging and Communications in Medicine) is the standard format used for storing and transmitting medical imaging data, but its complexity and metadata

structure makes it inappropriate for deep learning tasks. In contrast, NIfTI (Neuroimaging Informatics Technology Initiative) is a simpler, array-based format that represents volumetric data in a uniform structure. It is very compatible with image analysis libraries and deep learning frameworks like MONAI because it stores image data and spatial metadata (like voxel dimensions and orientation) in a single file. Therefore, for the optimal execution of this deep learning task it is necessary to convert the original imaging data from DICOM to NIfTI.

The dataset includes CT volumes stored as a series of DICOM slices and segmentation data stored as a single 3D DICOM Segmentation Object per patient. To prepare the data for modeling, both the CT volume and the segmentation file are converted to NIfTI. This process results in a 3D image volume and one or more binary segmentation masks, where all data is spatially aligned, which means that each voxel in the segmentation corresponds exactly to the same voxel in the CT image. The "Tumor" segmentation files for each patient were merged in one multi-tumor mask file, ensuring that there was an adequate number of tumor voxels in the segmentation file for each patient.

NIfTI files simplify the training process of CNNs by ensuring consistency in shape, orientation, and voxel spacing across all inputs. This alignment allows models to learn anatomical and pathological patterns without requiring complex preprocessing steps to resolve misalignments among volumes. The conversion pipeline ensures that every patient's image and segmentation data are output with identical dimensions, which is critical for 3D-image segmentation tasks.

6.4.2 Data split

The dataset was split into training, validation and test sets. The training set includes 70% of the original data and is used to optimize the model's parameters through back propagation. The validation set includes 10% of the original data and is used to monitor the generalization capabilities of the model on unseen data during training, after every epoch. The validation set's dice score is an important metric that determines checkpointing when the model's weights are progressing, early stopping when the dice is not improving and generally helps monitor overfitting. Lastly, the test set consists of 20% of the original data and is used only after the training process is complete to provide a final generalization estimate for the model, on completely unseen data during training. The dataset was randomly split into training, validation, and test sets once, before all training runs. The contents of each set were kept consistent across all experiments to ensure fair and accurate comparisons between different networks, hyperparameters, and training strategies.

6.4.3 Data visualization & insights

After the successful conversion of the input volumes, some valuable graphs and information was gathered for a better and complete understanding of the dataset and its specific needs. For these insights, only the train set was used. This way, data leakage for the characteristics of the validation and test sets is avoided, enhancing the model's robustness to overfitting. Moreover, these insights are quite valuable for the determination of the preprocessing strategy, which only involves the train set.

Firstly, the number of slices with tumor tissue was computed for each patient (volume), based on the corresponding tumor label masks. The top 10 patients have between

50 and 93 slices with tumor presence and therefore provide valuable volumetric data for the model, regarding the 3D shape and characteristics of tumors. In contrast, the bottom 10 patients only contain 2 to 4 slices with tumor presence and could therefore result to the domination of the background class. Approximately, 25% of the cohort contains less than 8 tumor slices, showcasing the class imbalance issues of the dataset.

Another important visualization was the distribution of axial slices among patient volumes. More specifically, Figure 6.2 shows a main cluster of volumes with depths around 30-60 slices and another one, less dense, between 120-170 slices. This level of shape variance presents a challenge for 3D segmentation with MONAI, as it best performs with uniform input shapes and structures. Therefore, the need for consistent resampling or cropping to standardize input shape is clear.

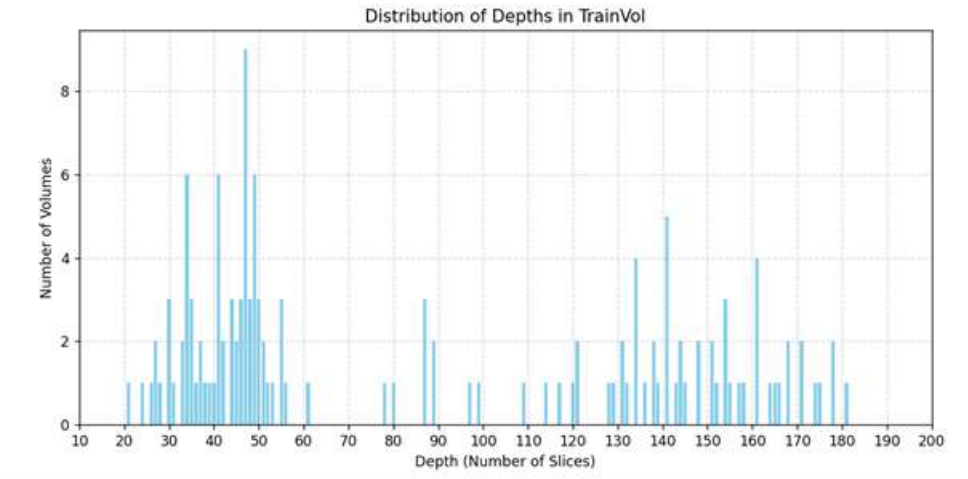

Figure 6.2: Depth distribution of train set volumes

Another valuable insight, based on the depth axis variety of the dataset, was the assessment of the percentage of slices that contain tumors, per patient. A random sample of 20 patients was selected for this task. The results showcase great variance, as some patients were found with 35-47%, while others contained tumor instances in only 4-8% of their total volume depth. These results verify the previous concerns for great dataset imbalance. However, an acceptable model performance can indicate great generalization potential, due to the increased variance of the data.

Overall, these analytics of the training set pinpoint important challenges in the dataset, such as the uneven slice depth, and strong tumor class imbalance. These findings are critical to determine an appropriate preprocessing strategy that can improve the performance of the model and decrease the potential of overfitting and loss of valuable data.

Optimized dataset

To handle the imbalance suggested above, we created an optimized data set that only contained patients with more than 10 slices containing tumor instances. We preferred this general approach instead of handpicking patients with poor labels. The split among train, validation and test sets, was again 70-10-20, with a total of 110 patients remaining from the initial cohort. This process was done to avoid using CTs that mainly include background and can therefore harm the performance of the model and increase time

consumption and memory usage without adding valuable information. The depth distribution of the optimized dataset is presented in Figure 6.3. It is evident that the mean depth value is increased and mainly patients with shallower scans were excluded. This allows for using deeper patches during training that contain more relevant tumor information for the CT, without the addition of extra padding for many scans.

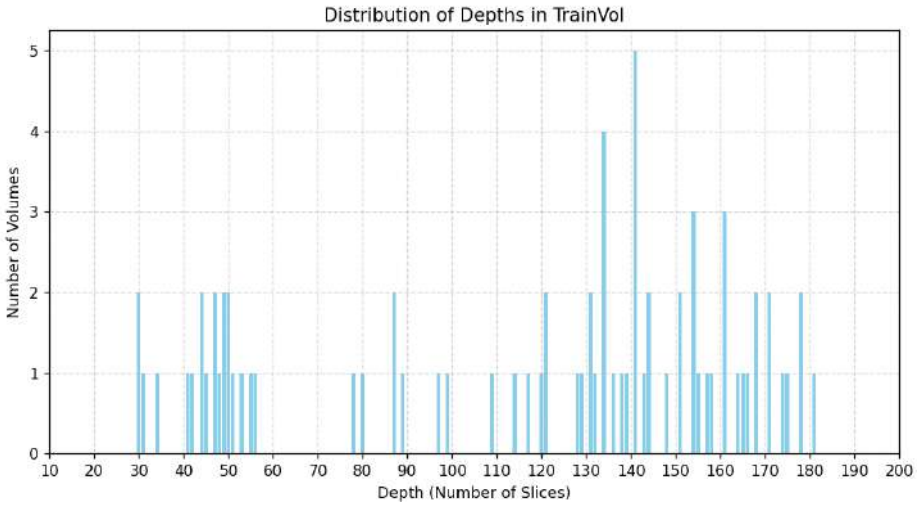

Figure 6.3: Depth distribution of train set volumes from the optimized dataset

6.5 Data preprocessing and augmentation

A crucial part of the training process followed was the application of thorough preprocessing and augmentation techniques. The augmentation transforms used were based on similar research work, as well as the unique requirements of this dataset, like the imbalance in tumor slices and the relatively small sample size (200 patients). They all derived from the MONAI library.

Before defining the transforms to be applied on the data, the deterministic seed for the deep learning task must be defined. In DL medical imaging tasks, the results should be consistent across multiple runs. By specifying a constant seed for the task, reproducibility is ensured. This means that the same code with the same inputs and settings gives the same result. This is crucial for moderating the effect of different changes on the strategy and pipeline and making debugging and comparisons easier. This process introduces a fixed random number generator for all random tasks of the pipeline, like data shuffling, weight initialization and data augmentation. As a result, the same exact transforms are reapplied to every patient in every epoch and even in different training runs, provided that other parameters like batch size and system configuration remain unchanged. The choice of the seed value as 0 is random, but aligns with similar experiments. Changing the seed number (ex. to 42 or 123) is sometimes useful for evaluating the robustness of results across different initializations. However, using a fixed seed ensures a clear implementation and robust comparison of different methods for the training process.

6.5.1 Basic transforms

Both the CT volumes and their matching segmentation masks underwent a series of preprocessing procedures to guarantee consistency and compatibility throughout the dataset.

These transformations from MONAI's dictionary-based transform pipeline standardize the data in terms of shape, orientation, spatial resolution, and intensity range, making the model training more effective. Together, these actions improve model generalization, lower training variance, and guarantee that the data satisfies the structural requirements of the specific network used.

Before the data is fed into the network, it is prepared and enhanced using a modular series of operations called the transform pipeline, which is applied to every sample in the dataset. To begin with, each volume with the corresponding liver and tumor masks is loaded from memory with "LoadImaged". The "EnsureChannelFirstD" changes the data shape from (H, W, D) to (C, H, W, D), since most deep learning models expect the channel to be the first dimension.

"SpacingD" transform is then used to resample the voxels of input images to a consistent physical size, along the x, y and z axis. This voxel spacing parameter is called pixdim. Table 6.1, contains the median voxel spacing values for each data split, along with the min and max values found in parenthesis. The height and width dimensions are almost isotropic in-plane, however the depth dimension displays higher variance. Based on these values, pixdim was set to (1.0, 1.0, 2.5). Especially for the z-axis, up-sampling from 5.0 to 2.5 improves the resolution and the volumetric detail being captured, without introducing extreme upsampling (up to 1.0), which could increase memory usage and introduce noise. This standardization can greatly improve the stability and generalization of the model. Moreover, the volume is interpolated in "bilinear" mode (effectively trilinear for 3D samples) for structural preservation during resampling. In contrast, the "nearest" mode is used for the segmentation to avoid generating artificial label values and ensure preservation the discrete class boundaries. CT scans across patients may be obtained in different orientations. With "Orientationd", all volumes are oriented to a specific coordinate system (RAS: Right, Anterior, Superior).

Table 6.1: Voxel Spacing Values

Axis	**Train Set (mm)**	**Validation Set (mm)**	**Test Set (mm)**
Height	0.79 (0.62 - 0.97)	0.75 (0.61 - 0.98)	0.78 (0.63 - 0.97)
Width	0.79 (0.62 - 0.97)	0.75 (0.61 - 0.98)	0.78 (0.63 - 0.97)
Depth	5.00 (1.5 - 7.5)	5.00 (1.5 - 7.5)	2.50 (0.8 - 7.5)

After that, "ScaleIntensityRanged" is used to normalize the intensity values of the input volume. The Hounsfield windowing process is necessary for this step. Relevant research suggests a window of -100 to 400 Hounsfield Units (HU) to best capture most lesions and soft tissues of interest around the liver area and thus exclude irrelevant areas like bones and air. After further visual inspection of the data using ITK-Snap as shown in Figure 6.4, the final window used during training was -100 to 200 HU.

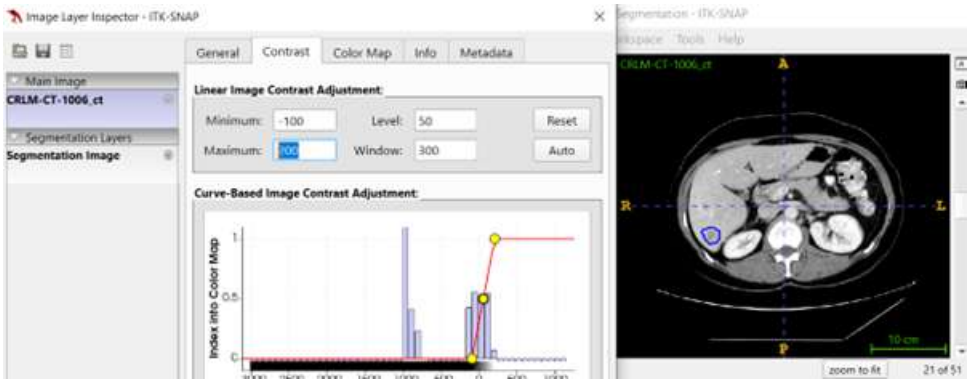

Figure 6.4: HU window inspection

Additionally, the HU intensity histogram on Figure 6.5 for a random patient of the dataset helps confirm whether the selected window covers the majority of meaningful voxel values. The transform is used to normalize this window to a range of [0.0 to 1.0]. The fine characteristics of the areas of interest are better visualized, and the model can focus on detecting the necessary tissue for successful tumor segmentation.

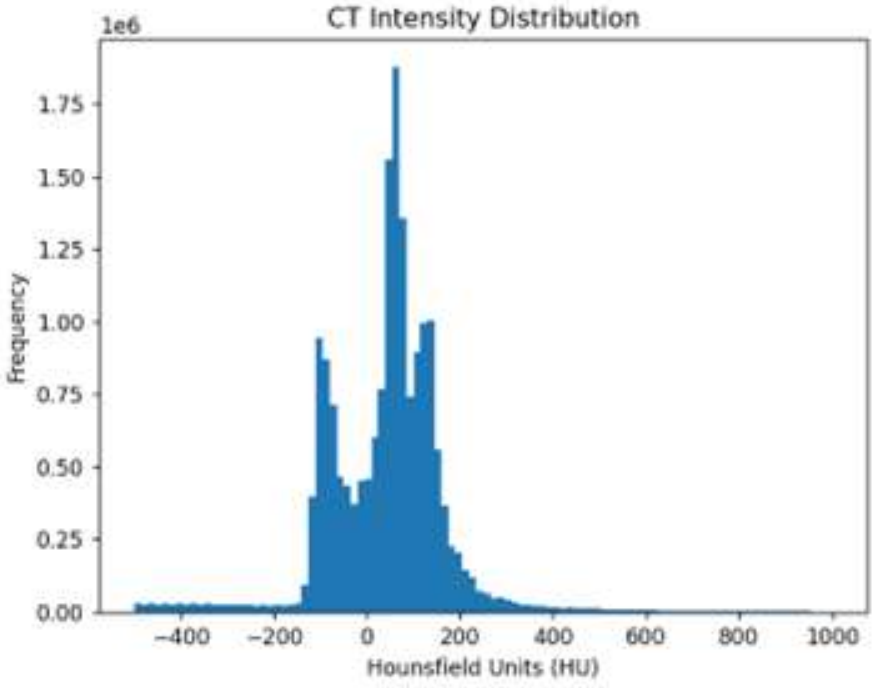

Figure 6.5: HU Intensity Histogram

The final result of the above basic transforms can be visualized in 6.6. It is obvious that the liver and tumor regions are much more visible after the augmentations, verifying their necessity.

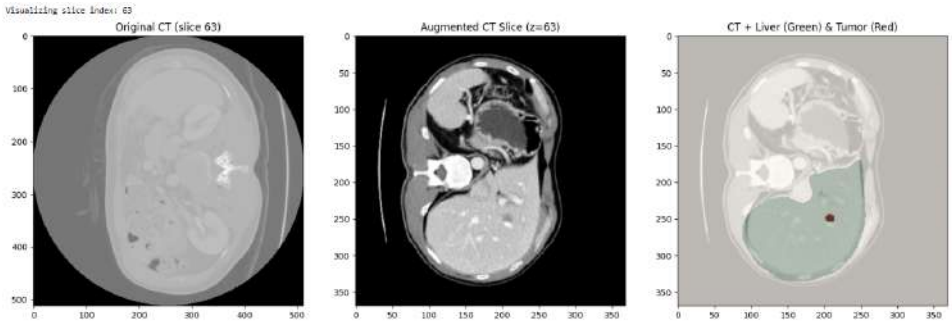

Figure 6.6: Original vs Augmented CT scans

6.5.2 Cropping

MONAI's "CropForegroundd" transform was used for cropping implementation. It crops the image to a tight bounding box around the defined source key, which was the liver label. That way the focus on the main ROI instead of the whole CT was ensured. Each patient's liver size differs, leading to further dimension errors like before. The "DivisiblePadD" transform is then introduced to make all crops divisible by k = 16 (or k=32 for transformer networks) with padding, ensuring a smooth downsampling and upsampling process during the network training.

All the above fundamental preprocessing transforms are consistently applied to all data splits, for both liver and CRLM segmentation tasks, to ensure a uniform input format for the model. This way, the performance of the model can be objectively evaluated, without preprocessing inconsistencies affecting it.

6.5.3 Augmentative transforms

The above transforms were adequate for liver segmentation. However, for the tumor task, the model still underperformed, mainly due to the strong imbalance both in the number of tumor voxels and in the depth (z-axis) of the initial volumes across patients. They are important for tumor segmentation tasks on imbalanced and small datasets like the one used, as they ensure sufficient generalization, prevent overfitting and help the model better understand tumor characteristics. They are only applied on the training set, because if applied to the validation or test sets, they would lead to distorted input distribution and biased performance evaluation.

Introducing the "RandCropByPosNegLabeld" transform from MONAI into the pipeline was one of the main propositions of this project and a crucial step in addressing the significant data imbalance of this task. Some scans may only have a few slices with tumor presence, while others have substantial tumor instances. 'RandCropByPosNegLabeld' helps mitigate this problem by generating balanced training samples through controlled random cropping. This transform selects a specified number of random sub-volumes from each scan, with a ratio of patches that include positive tumor voxels (pos) and those that do not (neg). By doing so, it ensures that the network is exposed to a meaningful and diverse distribution of tumor-containing regions, which is critical when tumors occupy only a small fraction of the volume. Moreover, the number of positive tumor samples is effectively increased, without the need to discard slices with low tumor presence, an approach that would otherwise risk creating a biased and less realistic dataset prone to overfitting. Without this strategy, the network would be biased towards learning back-

ground features, as it would mainly see non-tumor regions, especially in patients with small or few tumors 6.1.

The initialization of the transform and its parameters can be seen in Listing 6.1. By setting label_key=”seg_tumor”, we ensure that the model uses the tumor label to define ”positive” samples. Based on the transform’s implementation, the model finds all foreground voxels for each sample (in this case tumor voxels), selects a center near a tumor voxel and creates a random crop around this center based on the defined spatial size.

Listing 6.1: Cropping augmentation configuration

```
RandCropByPosNegLabeld(
            keys=["vol", "seg_tumor"],
            label_key="seg_tumor",   # crop around the liver or
               tumor
            spatial_size=(128, 128, 48), # Slightly smaller due to
               smaller tumors
            pos=1, # With probability pos/(pos+neg), it chooses a
               center inside the liver region
            neg=0.2,
            num_samples=6,
            image_key="vol",
            allow_smaller=True
),
```

The spatial size depth was carefully selected based on the depth imbalance noticed in the initial dataset. In cases where the depth requested is greater than the actual, MONAI introduces padding to the volume to meet the requested depth. This could result in the return of duplicated or padded regions, worsening the sample’s quality. In contrast, when the requested depth is less than the actual one, the number of valid positions of the positive tumor centers increases and therefore the quality of the sample is not affected. We experimented with various values ranging from (96, 96, 32) for heavier models to (160, 160, 64) for lighter ones. By cropping to a fixed spatial size, batching and training stability issues are permanently addressed.

The transform does not resize or shrink the image, since this could damage the resolution and shape of the tumor. Instead, it just selects a subregion containing tumor voxels with probability:

$$\Pr = \frac{\text{pos}}{\text{pos} + \text{neg}}$$

For each patient in the DataLoader, “num_samples” defines the number of samples generated by the transform. The tumor label files for each patient were combined, to make sure that for each patient there was an adequate number of tumor voxels.

This transform was preferred from other spacing transforms, to maintain the best possible image quality. For example, “ResizeD” transform, which essentially is a uniform interpolation, performs the resizing of the CT by introducing blur, which could alter fine details in small tumors. However, it was used for liver segmentation, as the size and shape of the ROI is more consistent and clearly defined.

6.5.4 Random augmentations

Random augmentation transforms introduce variability in both the spatial and intensity aspect, enhancing model generalization. Each transform is applied to the sample based on the specified probability. To begin with, RandFlipd randomly flips the image along the specified axis to help the model become indifferent to anatomical orientation. RandRotated introduces rotations around the x, y and z axis, similar to possible misalignments occurring during CT scan procedures. RandZoomd encourages robustness to organ size differences and scale variations among patients, by zooming in and out of the volume.

RandScaleIntensityd slightly adjusts brightness, while RandShiftIntensityd offsets voxel values to simulate scanner calibration differences. RandGaussianNoised introduces low-level noise, resembling imaging artifacts, and RandAdjustContrastd adjusts contrast non-linearly, strengthening the model's ability to handle different tissue visibility levels.

In general, each patient sample goes through all transforms sequentially. It starts from the fundamental transforms which are mandatory and moves on to "RandCropByPosNegLabeld", which is a generative transform that creates multiple patches for each sample. The rest of the stochastic augmentative transforms are applied to each generated patch based on the independent probabilities of each one. They are applied every time a new sample from the dataset is fetched, even if it is cached, preserving data augmentation during training.

Figure 6.7 presents the result of the complete preprocessing pipeline for both the volume and the tumor mask. It is evident, that the volumes are now much more prepared for the tumor segmentation task.

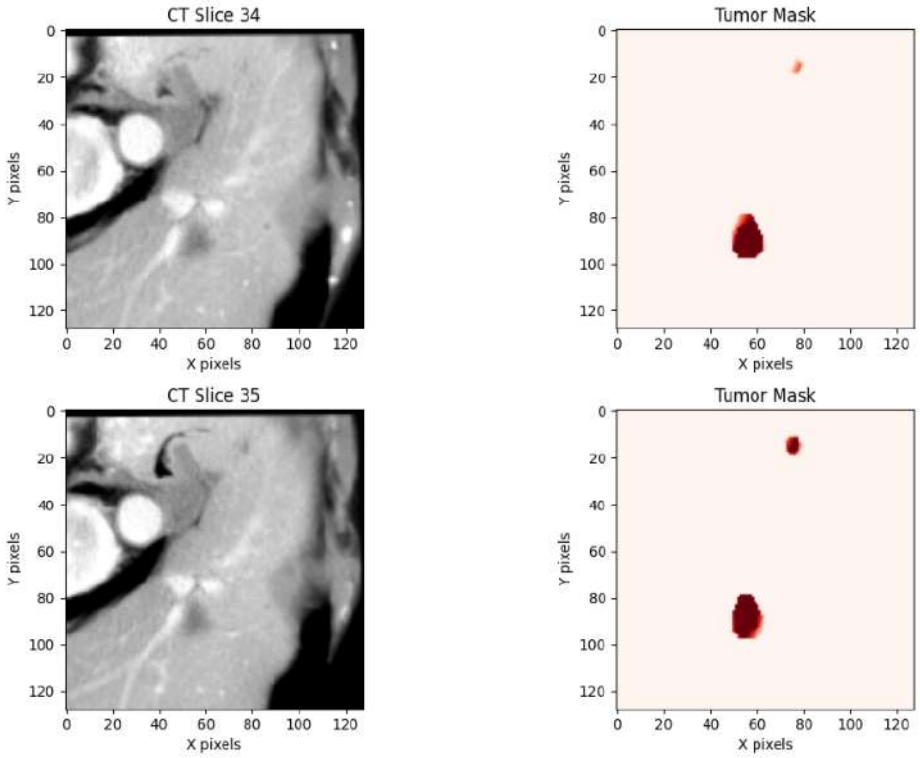

Figure 6.7: Example of complete preprocessing result

6.6 Model architectures

Various model architectures, imported from MONAI, were used during experimentation to compare model performance and identify the most suitable architecture for this specific segmentation task. The following section briefly outlines the key models, their main parameter choices, and their distinctive characteristics. Detailed implementation code for each network is provided in Appendix 9.2.

All models are configured with:

- One input channel representing the CT volumes, during initial experiments and two input channels for the final automated pipeline.
- Two output channels representing background and foreground classes (liver or tumors).
- Spatial dimensions set to 3, since the data are 3D volumes.
- A small dropout rate for regularization and overfitting control.
- Input spatial sizes divisible by 16 for compatibility with transformer-based networks.

Model Summaries

- **UNETR:** Utilizes a transformer as the encoder to capture global dependencies across the input volume. A *multi-head self-attention* mechanism enhances representation learning. Skip connections link encoder and decoder for better gradient flow. *(Implementation in Listing 1)*
- **Swin UNETR:** Incorporates a hierarchical Swin Transformer backbone with shifted windows, enabling efficient local-global context modeling with reduced computational cost. Multiple resolutions are used to progressively exchange information between windows. The default layer dimensions and normalization settings are used. *(Implementation in Listing 2)*
- **Attention U-Net (3D):** A 3D adaptation of the classical U-Net with attention gates at skip connections to emphasize relevant features while suppressing less informative regions. Channel depth and stride choices balance feature capacity and memory efficiency. *(Implementation in Listing 3)*
- **U-Net with Residual Units (ResUNet):** Standard U-Net architecture enhanced with residual units at every layer to improve optimization and gradient flow. Provides strong baseline performance with fast training, ideal for testing different training configurations. Various channel depths are tried, to find the best balance between feature learning capacity and memory usage. The rest of the configurations are set to the default values. The default normalization normalizes each image instance independently which is better for inputs with significant variations in intensity, contrast, and noise. *(Implementation in Listing 4)*
- **SegResNet:** Higher initial filter counts help capture subtle lesions without excessive memory use. Residual blocks are implemented throughout the encoder/decoder. Dropout is also used, along with the default settings for normalization

and number of upsampling and downsampling blocks. Lastly,the default 'GROUP' normalization divides the channels into groups and normalizes within each group, making its performance independent from batch size. *(Implementation in Listing 5)*

6.7 Training strategy and hyperparameters

The training process follows a conventional supervised learning strategy using a custom training loop implemented using PyTorch and MONAI. The preprocessing and training code was based on an open-source implementation available on GitHub [108], and then adjusted to the specific needs of this project. The pipeline was split into three main Python files, preprocessing (which was covered previously in this chapter), training and utilities.

6.7.1 Utilities script

The utilities file contains helper functions necessary for the training process, as well as the main train loop. Training is performed over multiple epochs using mini-batches, and model performance is monitored using a combination of loss values and segmentation quality metrics.

Metrics and transforms setup

Firstly, the necessary metrics and transforms to be used inside the train loop are declared. We use the MONAI **'DiceMetric'** implementation, which is a common practice in similar projects. This metric calculates the dice similarity coefficient (DSC) between GT and predicted masks for each patient, averaged across each epoch. It is a voxel-wise implementation that calculates the number of positive voxels that belong to the intersection of GT and predicted mask sets and then divides this with the sum of positive voxels that belong to each set.

We exclude background from the calculation to ignore the dominant background channel and shift the model's focus on capturing the tumor regions. The metric processes predictions and labels as discrete masks instead of their original raw logit form. That is why the predictions go through a post-processing transform that produces a single-channel label map with the most likely class for each voxel through 'argmax'. Then the label map is expanded into a two-channel one-hot encoding, with each voxel's class represented by a binary vector. The GT masks are also one-hot encoded to fit the same format. This setup ensures that the dice computation is accurate and does not include soft predictions, which are raw low-confidence softmax probabilities that can inflate the final DSC result.

In general, including the background class in the "DiceMetric" calculation can inflate the final value, since the background covers the majority of the volume and is easily segmented. Therefore, it is a good practise to only include the foreground class to compute the DiceMetric for 3D segmentation tasks, so that the results are an accurate reflection of the model's ability to identify the ROI. However, as discussed in Chapter 5, some papers include background in the computation. For this project, we are going to present the results from both methods to showcase the consequences of each approach.

Additionally, MONAI's 'Surface Distance Metric' is initialized along with a 'LargestConnectedComponent' tool that removed small FP predictions from ASSD calculations.

Train loop

The main training loop follows, which moves the volumes and masks from the train set to the GPU to begin the process. The automatic mixed precision (AMP) method is used to accelerate training and reduce memory usage during large matrix computations. Additionally, the 'GradScaler' from MONAI is used to prevent underflow, by multiplying the network losses by a scale factor, so that gradient values are not truncated to zero.

Then, discrete dice computation takes place, where batched tensors are divided into per-sample tensors and the post-processing transforms are applied to each sample to create the final discrete prediction and GT values for 'DiceMetric'. At the end of each epoch the mean DSC and loss over all training samples is calculated.

Validation loop

The validation loop follows the same overall structure with some extra functionalities.

During the training of the validation set, sliding window inference is used. This module divides the large 3D volumes in smaller overlapping patches to avoid memory issues while preserving full-volume evaluation during the training process. For small and sparse tumor, a bigger overlap value 0.5-0.7 is suggested to make sure that instances of the same tumor but in different slices are more likely to be on the same patch. It's necessary because validation data is neither cropped like training data during preprocessing, nor resized to avoid possible blur or noise and therefore can overload the GPU. The region of interest shape is set to match the training set's patches because a smaller size can lead to valuable context being lost. After each sub-volume is processed, the predictions are combined back together to form the final full-volume output.

Additionally, key performance metrics are calculated based on the predictions on the validation set for each epoch. Based on relevant research in tumor segmentation tasks, False Negatives are quite important, since they reflect missed tumor instances by the model. To monitor this issue, the Recall metric, also known as sensitivity, was incorporated to measure the actual positive lesions found. Similarly, the precision metric highlights the number of incorrectly predicted lesions. Lastly, the ASSD metric is recorded, to provide an extra indicator of the anatomical correctness of the predicted masks compared to the GT, after the exclusion of small FP and empty pairs.

Dice-Focal loss function is used as the primary training and validation loss because it combines DSC accuracy with a focus on ambiguous voxels, making it ideal for tumor segmentation tasks with high imbalance. Train loss guides the model's learning by updating weights to minimize error, while validation loss checks how well the model generalizes to unseen data. Dice and loss values are aggregated with the same process followed during the train loop. However, the global dice is also computed, to highlight the performance of the model on larger lesions. TP, FP and FN values are accumulated inside the loop for every batch and aggregated after the epoch is over, for the global dice calculation.

Model checkpoints are saved based on the best validation Dice score, and early stopping is triggered when no improvement is observed after a predefined number of epochs. This balances training efficiency with performance, helping avoid overfitting.

After the train and validation loops are completed, and the best model is saved, it is set to evaluation mode to be used for predictions on unseen test data. The test data go through the same inference and AMP processes, and the DSC, global DSC, precision and recall metrics are calculated to be compared with the corresponding results obtained during training.

6.7.2 Training script

The training script includes the complete experimental training pipeline. It begins with the option to either load preprocessed data from a local cache or repeat preprocessing from scratch. This flexible setup supports and faster model experimentation and a clear understanding of each run.

Following data preparation, the script continues with the definition and configuration of key training components and their relevant parameters. More specifically, the loss function and the neural network architecture are initialized, followed by the model with the corresponding Adam optimizer that contains important hyperparameters. The loss function calculation includes the background class, but with a 1/10 weighting ratio in favor of the minority class during the dice loss computation, to increase the penalty for misclassified tumors and improve the model's performance.The *train()* function is then called with all the parameters required to initiate the training and validation phases over a specified number of epochs.

Hyperparameters

In DL tasks, the hyperparameters are typically set before each training process begins and control crucial aspects of the learning process. They influence the model's performance, complexity and learning ability. The main hyperparameters used in this task, along with their selected values are described below.

- The **learning rate** chosen for the best-performing model was 1e-4, a value commonly adopted in related research. In primary experiments for hyperparameter selection, the same U-Net architecture was used with different learning rates. It was observed that 1e-3 led to rapid and premature convergence, before sufficient generalization of the model. On the other hand, a lower value of 1e-5 resulted in very slow training progress and required significantly more epochs to reach comparable performance. Based on these observations, 1e-4 was deemed the appropriate value to balance training stability and convergence speed.

- The **batch size** refers to the number of training examples used in each iteration of the optimization algorithm. The value selected for most experiments is 1, due to the increased size of the 3D CT scans and the limited GPU capabilities.

- The number of **training epochs** selected was 200-300 with an early-stopping mechanism after 20 epochs to save time and memory usage.

- **Weight decay** is a regularization technique that helps avoid overfitting in deep learning tasks such as tumor segmentation. It adds a penalty proportional to the squared magnitude of the weights to the loss function. The chosen value for this task was $1e^{-5}$, which is a widely used initial value that offers a little regularization without limiting the learning capacity of the model.

- **Dropout** is another regularization technique, which defines the percentage of randomly selected neurons to be ignored during training, aiming to prevent overfitting. The usual value selected was 0.1, indicating that only 10% of the neurons in a given layer will be set to zero at each training step. A low dropout value is chosen to increase the model's capacity to learn the subtle tumor features.

- **Activation functions** determine the output of a neural network node given a set of inputs. During network initialization, ReLU was commonly used as the default option for our experiments. However, Leaky ReLU and PRELU (which is the default option for UNet) were also used. These functions have learned a small slope for negative inputs, which is parametric in the case of PRELU. For the loss function both sigmoid and softmax were used, but the final choice was softmax, since the segmentation channels are mutually exclusive and this function assigns a single class per voxel.

- A **scheduler** is also selected to facilitate learning rate changes during training. More specifically the 'ReduceLROnPlateau' scheduler from MONAI is used after the validation metrics have been calculated to reduce the learning rate in case validation DSC stalls, helping the model capture finer details.

6.7.3 Testing script

A notebook was used to run an explicit evaluation process that calculated the testing metrics again and contained visualizations from the predicted masks and other useful plots that indicate the accuracy and strengths of the model.

After loading the best weights of the model and setting it in evaluation mode, we used 'DiceMetric', without including the background, to compute the DSC and other metrics on unseen data during training. The final mean values for each metric are presented along with the standard deviation, minimum and maximum values found. This is an important step that proves the model's generalization ability, which is crucial for its potential future application in real-time medical data.

We added another chart that displayed the distribution of per-patient DSC across 5 percentage ranges (0–20, 20–40, ..., 80–100). The main purpose was to examine potential outliers that decrease the overall results and also evaluate the dataset optimization strategy that was applied. We also visualize the GT mask next to the prediction contour, which is the outline of the predicted region. We implement that for two patients from the top bin and two patients from the bottom one to visualize differences in GT labels, image quality and potential weaknesses of our model. These visualization also provide insights on the overall accuracy of the predicted regions and the characteristics of possible false positives or missed tumor instances.

Lastly, the original scans, ground truth labels and prediction labels are printed for some random patients for thorough visualization of the quality of predictions. The outline of the tumor instances are included for GT and prediction slices to showcase the accuracy of the predicted regions and possible false positives or missed tumor instances.

6.7.4 Fine-Tuning

Another experiment was conducted, this time involving the fine-tuning of a pretrained model. A 3D segmentation model trained for spleen delineation from CT images was selected from MONAI ZOO for this transfer learning task. The model was trained on the MSD dataset, processing $96 \times 96 \times 96$ pixel patches, with the U-Net architecture. This model was selected due to its similarities with the U-Net model implemented for our task, since successful fine-tuning requires identical dimensionality, layer shapes and depths, input channels and normalization. The U-Net structure was slightly edited to

better fit the pretrained model. The activation function was switched to 'PRELU', the feature normalization type was set to 'BATCH' and dropout was reset to 0.

The model was downloaded from MONAI ZOO and the path to the checkpoint file "model.pt" was built for weight extraction during our model initialization. After that, a function was created with the purpose to load the pretrained weights. After loading all the weights, the final layer or head of the model needs to be re-initialized because it maps spleen-specific features. In U-Net architecture the head is usually a Conv3d layer with as many output channels as the number of classes. We visualized our U-Net structure and saw that the last layer had two output channels and a $3 \times 3 \times 3$ kernel. Therefore, the last Conv3d layer of the model was selected and re-initialized. This process is done on CPU and then the model is moved to GPU to begin training. A new optimizer is created, with a greater learning rate for the fresh head, so that it learns the new tumor features faster.

Additionally, since a pretrained SegResNet network that matched our purpose, was not found, we decided to try and pretrain the model on our own data only for liver segmentation for 100 epochs. The weights were then used to fine tune the same model architecture for the tumor segmentation task.

6.7.5 Combined Pipeline

After the best models for liver and tumor segmentation are selected, they are combined to form an automated segmentation pipeline. More specifically, we use the whole dataset as the test set, and create liver predictions for the entire dataset with the best liver model. Then the transforms used are inverted with the 'Invertd' transform from MONAI, which undoes preprocessing and reverts the predictions back to the original image space. 'AsDiscreted' converts the prediction logits into a hard label map by taking the argmax class at each voxel and then predictions are saved under the specified folder and name with 'SaveImaged'.

The predictions saved are then used during the preprocessing stage of the tumor segmentation task. They replace the original liver GT for all preprocessing transforms. For example, the volume is now cropped around the predicted liver mask and not the original GT. Moreover, the predicted masks are added as a second input channel with 'ConcatItemsd' transform, which stacks the volumes and liver predictions together, so that they are loaded together during training. The mask channel acts as an extra mechanism that shifts the network's focus towards the liver region, when searching to identify tumors. The CT channel includes important appearance features, while the liver channel provides spatial information.

Chapter 7

Experimental Results

In this chapter, we are going to present the results gathered from the different trials and experiments mentioned previously. For this purpose, metric curves and plots are presented for better visualization of the training and testing performance across different training parameters and model architectures. The predicted masks and the corresponding ground truth labels are also compared for both liver and tumor segmentation, along with other valuable evaluation insights. These results aim to describe the performance of the proposed methods and models and indicate which areas could be further improved.

As we discussed previously, the W&B framework was leveraged to provide metric visualizations during the training phase of the model that give valuable insight into its performance. The curves from various experiments are presented and compared in this section.

7.1 Liver segmentation

7.1.1 Architectures comparison

SegResNet was pretrained on liver to later be fine-tuned, however showed great potential. Additional experiments were conducted with UNETR and ResUNet. For all experiments the original dataset was used. The training characteristics are shown in Table 7.1.

Table 7.1: Basic configuration by architecture.

	ResUNet	**UNETR**	**SegResNet**
Max Epochs	200	200	200
Learning Rate	0.0001	0.0001	0.0001
Patience	20	20	20
Activation	softmax	softmax	softmax
Loss	DiceFocal	DiceFocal	DiceFocal
Hardware	RTX 4080 16GB	RTX 4080 16GB	GTX 1650 4GB
Spatial Size (H-W-D)	128-128-48	160-160-64	128-128-48

The final training results are displayed in Table 7.2

Table 7.2: Performance results across architectures

Metric	ResUNet	UNETR	SegResNet
Epochs	70	53	78
Train Dice	0.983	0.980	0.980
Train Loss	0.195	0.022	0.02
Final Val Dice	0.960	0.948	0.966
Final Val Loss	0.037	0.047	0.031
Final Val Recall	0.953	0.948	0.954
Final Val Precision	0.956	0.936	0.097
Final Val Surface	0.613	0.745	0.534
Test Precision	0.954	0.951	0.96
Test Recall	0.957	0.095	0.964
Best Val Dice	0.962	0.952	**0.969**
Test Dice Softmax	0.963	0.956	0.968

As we can see, the best performing model was the SegResNet. All results were close on most metrics, proving the robustness of the initial setup for the training process. Additionally, the test set results verify the lack of overfitting and the substantial generalization capabilities of all models. Lastly, the validation metric curves can be visualized in Figure 7.1

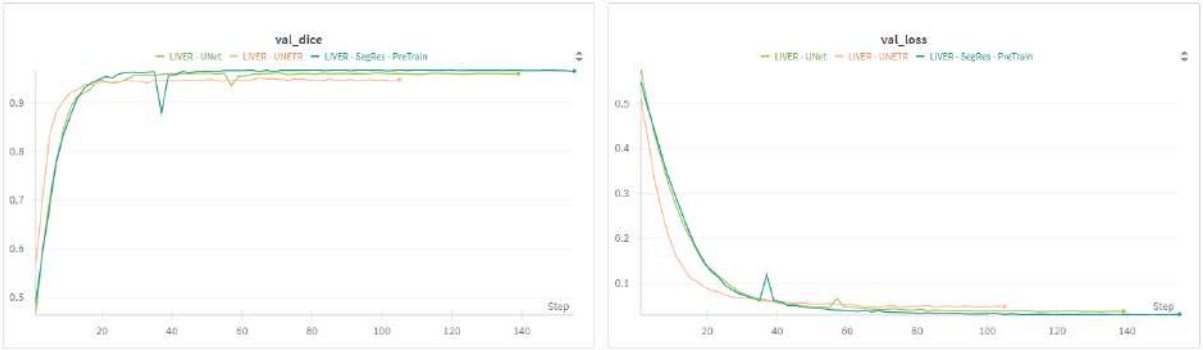

Figure 7.1: Validation curves for Liver Segmentation comparison

7.1.2 Qualitative evaluation

The best performing model was used for further evaluation on unseen data. Table 7.3 presents the final odel results

Table 7.3: Testing results for SegResNet on 41 test volumes

Metric	Mean	Std	Min	Max
Dice	0.9681	± 0.0293	0.8443	0.9788
ASSD	1.0058	± 0.6580	0.3437	3.8343
Recall	0.8879	± 0.0739	0.6368	0.9721
Precision	0.9672	± 0.0244	0.8553	0.9949

An example of liver predictions by this network is shown in Figure 7.2. The great similarity between the GT and the predicted masks is evident in this figure, verifying the high-level performance of the model.

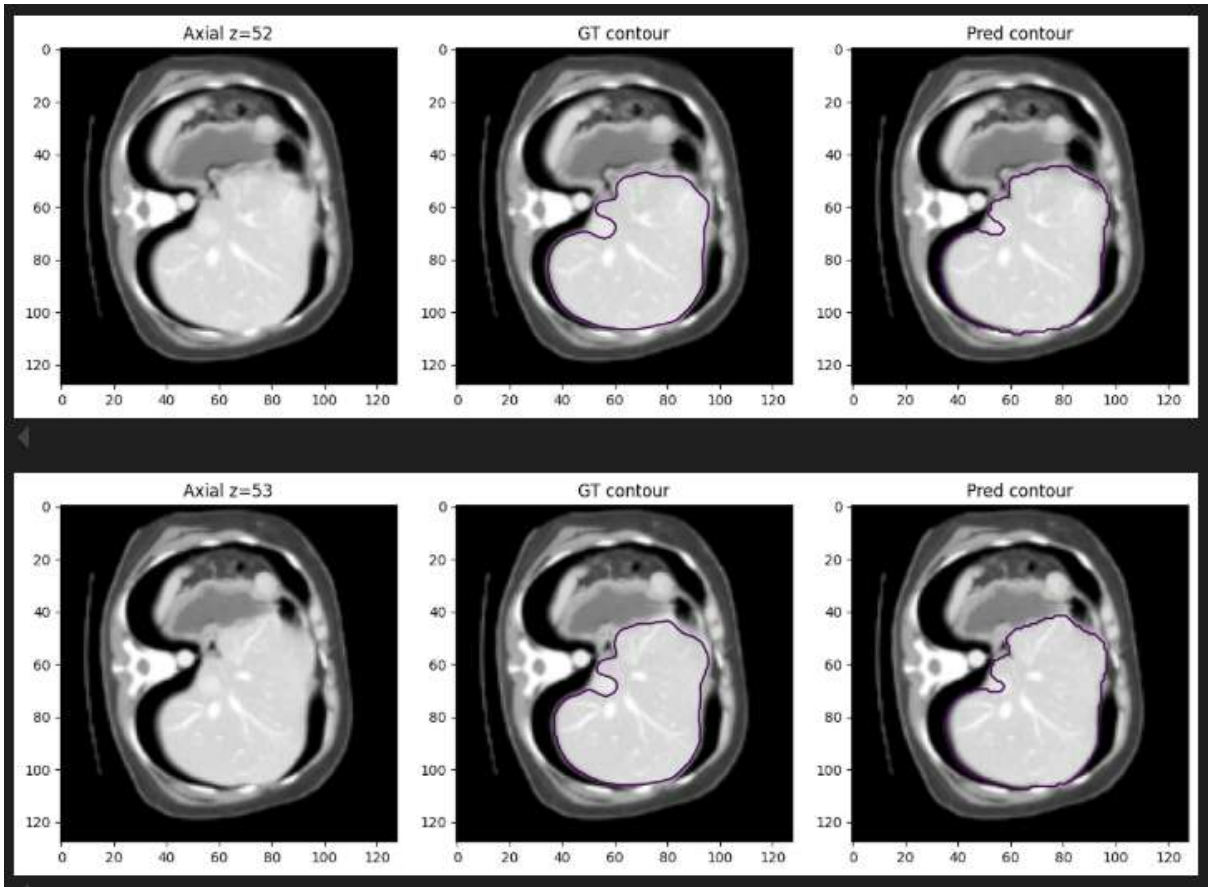

Figure 7.2: Liver prediction masks

7.2 Tumor segmentation

7.2.1 Basic trials

The initial trials involved the comparison of similar models in order to determine some fundamental settings for the task, regarding metric calculations, data preparation and hardware selection, before moving on to the comparison of different architectures and parameters.

Background inclusion

Figure 7.3, contains the comparison of 2 different runs with the U-Net architecture and the same hyperparameters and preprocessing techniques. However, the pink graph represents a run, where the background was included in the dice metric computation. That explains the gap between the two curves in the dice metric graphs, since the inclusion of background provides a significant head-start to the model's dice, since it is far easier to be correctly segmented. In contrast, with the background excluded, the dice values begin almost from 0, until the model slowly and gradually learns the tumor-specific features and is finally improved.

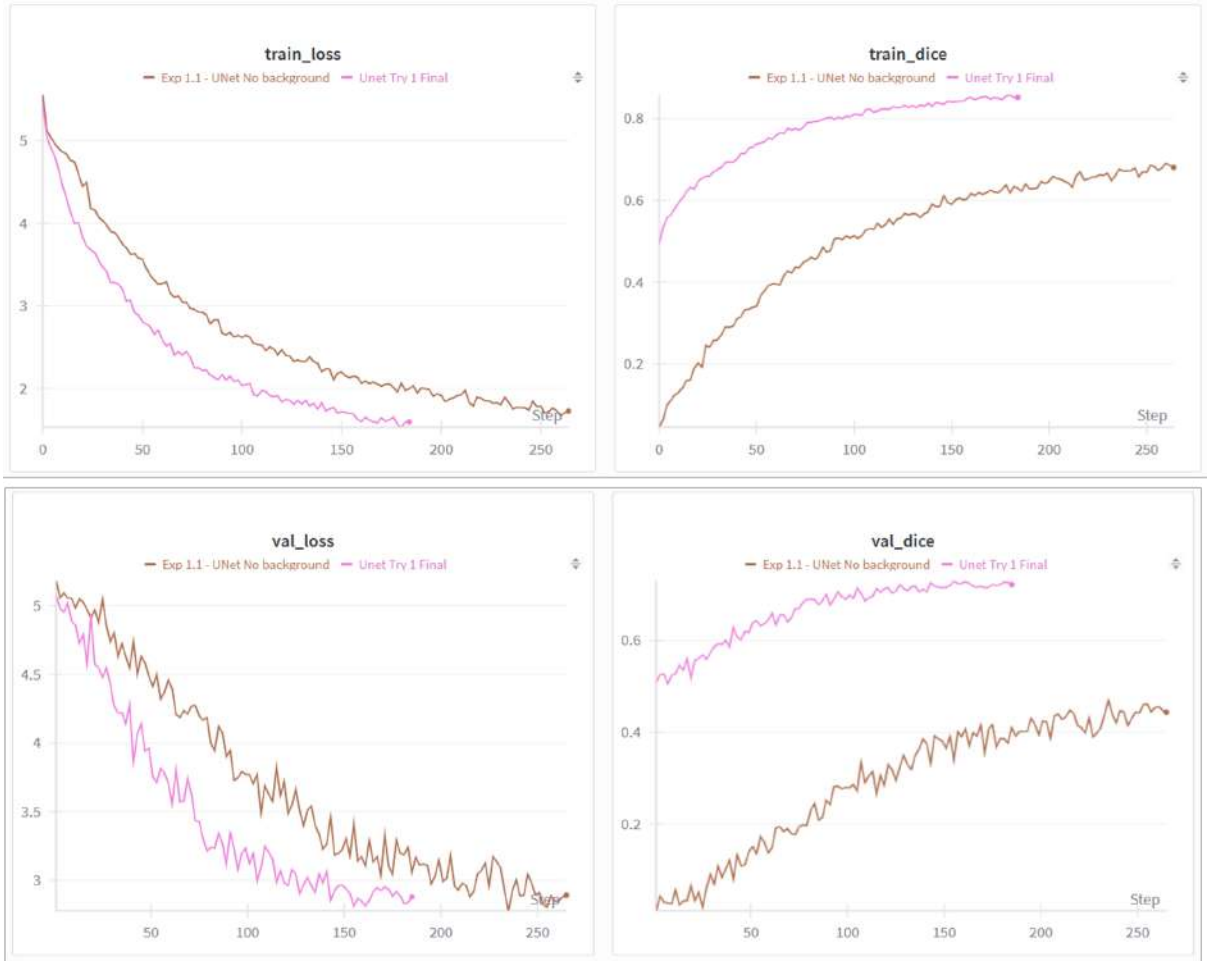

Figure 7.3: Comparison for background inclusion

Dataset optimization

Figure 7.4 compares two identical UNet implementations, with the only difference that in 'Exp 1.5' an optimized instance of the initial dataset was used as input, which only contained patients with more than 10 slices that contained tumors, for all 3 dataset splits. It is obvious that the optimized strategy improves the performance of the model, since the final dice and loss values are improved, while the validation metrics converge faster, meaning that the model can understand the features easier now that it is not fed with CTs where background is too dominant.

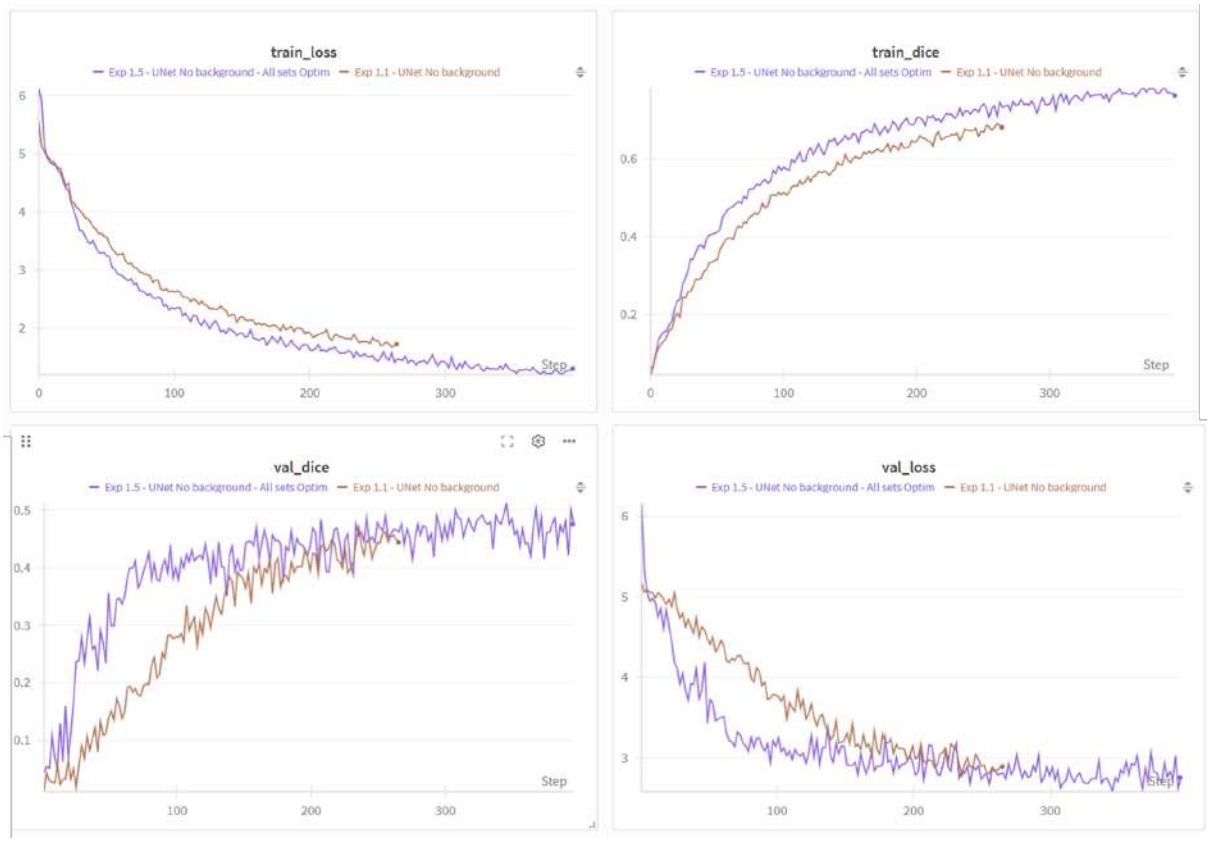

Figure 7.4: Comparison for dataset optimization

Hardware comparison

We compared the training log instances from experiments on the NVIDIA GEFORCE RTX 4080 16GB GPU and the local GEFORCE GTX 1650 4GB respectively, for the same SegResNet architecture and similar spatial sizes. It was observed that there is an immense difference in the runtime speed of the enhanced GPU, as each epoch lasted around 57s, while the local GPU took around 1300s. This significant reduction in training time highlights the substantial impact of high-end GPU hardware on model training efficiency. Faster training not only accelerates experimentation but also enables the use of larger batch sizes and more complex models without excessive runtime overhead.

7.2.2 Architecture comparison

After finalizing the main hyperparameters and strategies to be used for the training process we move on with the comparison of different architectures. Four of the most commonly used architectures were tested, SwinUNETR, ResUNet, AttentionUNet and SegResNet. The basic parameters and configurations for the training process that were common for all architectures are displayed in Table 7.4.

Table 7.4: Training configuration used for CRLM segmentation experiments.

Parameter	**Value**
Activation Function	softmax
Max Epochs	300
Loss Function	DiceFocalLoss
Dataset	Optimized
Hardware	GeForce RTX 16GB
Loss Weights	1/10
Patience	40
Evaluation Metric	DiceMetric
Learning Rate	1×10^{-4}
Sliding Window Overlap	0.4

Table 7.5 contains the different spatial configurations implemented to ensure that the models fit the memory and did not result in "OutOfMemory" errors due to increased size. Each training sample is an additional 3D patch loaded into the GPU, hence many samples can exceed the available VRAM. The final epochs indicate the point where early stopping was triggered for each model. The earlier the final epoch, the earlier the model reached a plateau and stopped improving on the validation data score.

Table 7.5: Differences in training configurations across architectures.

Parameter	**SwinUNETR**	**ResUNet**	**AttentionUNet**	**SegResNet**
Spatial Size	96, 96, 64	128, 128, 64	112, 112, 64	128, 128, 64
Number of Samples	5	12	8	8
Final Epoch	102	107	220	171

The final results on the unseen testing set, as well as the best validation dice reported, are presented in Table 7.6.

Table 7.6: Final results

Parameter	**SwinUNETR**	**ResUNet**	**AttentionUNet**	**SegResNet**
Test Dice	0.525	0.604	0.641	**0.652**
Test Precision	0.508	0.596	**0.665**	0.614
Test Recall	0.624	0.647	0.694	**0.75**
BEST VAL DICE	0.492	0.596	0.637	**0.646**
Test Global Dice	0.596	0.609	**0.673**	0.633

Lastly, the training curves for dice and loss during validation, as well as corresponding metric curves for the four models are shown in Figures 7.5 and 7.6. These graphs provide an analytical view of the training process, highlighting convergence behavior and potential signs of overfitting.

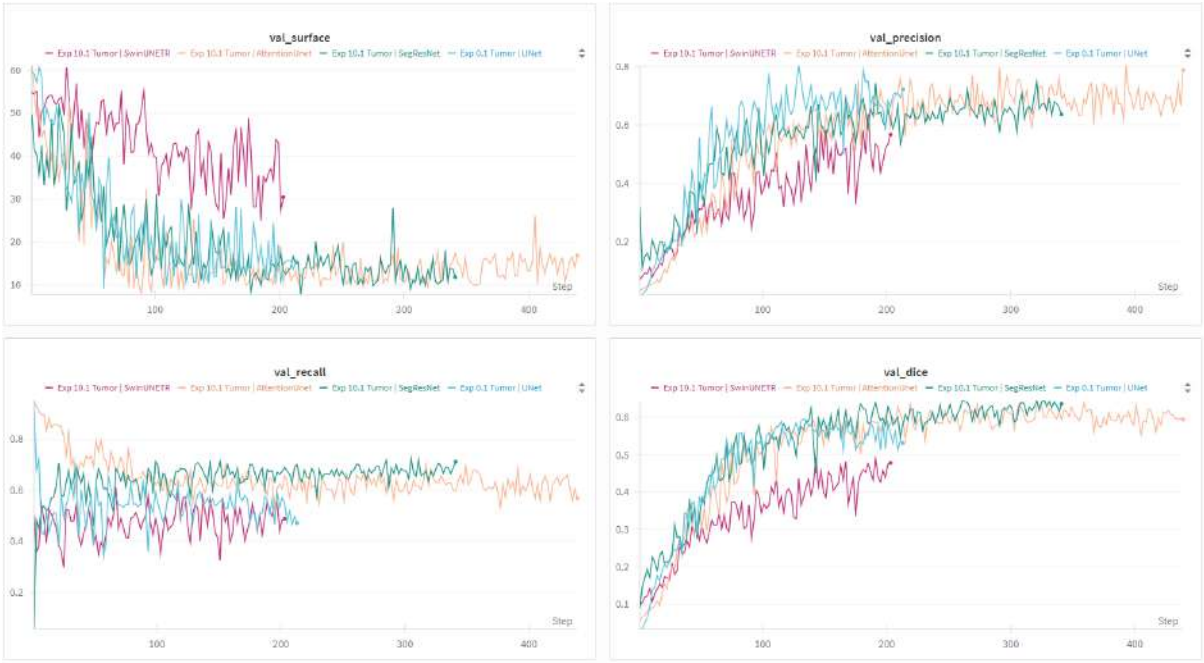

Figure 7.5: Metric curves

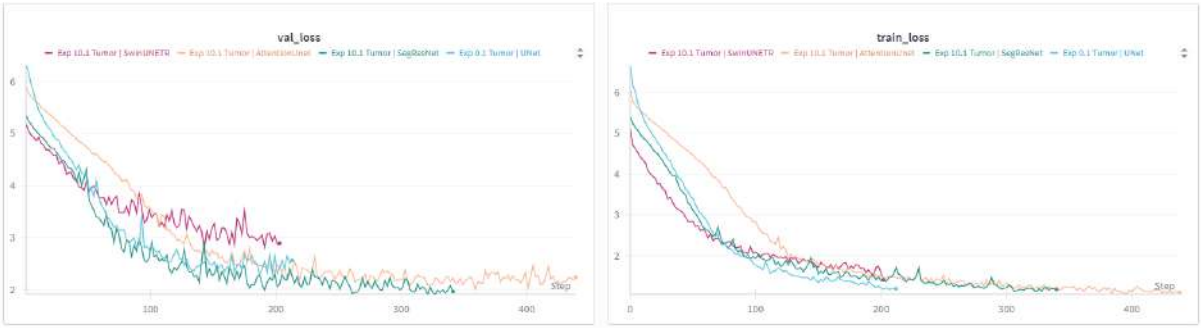

Figure 7.6: Loss curves

Loss comparison

We used the best model (SegResNet) to conduct an experiments between Dice and Focal Loss (green) and Dice and Cross-Entropy Loss (grey). The other specifications were exactly the same. The results are shown in Table 7.7 and Figures 7.8, 7.7. We can see that the results are similar, however DiceFocal performs slightly better in dice metrics and precision, while keping recall at a satisfying level.

Table 7.7: Final results

Parameter	**DiceCE**	**DiceFocal**
Test Dice	0.63	**0.652**
Test Precision	0.582	**0.614**
Test Recall	**0.761**	0.75
BEST VAL DICE	0.624	**0.646**
Test Global Dice	0.604	**0.633**

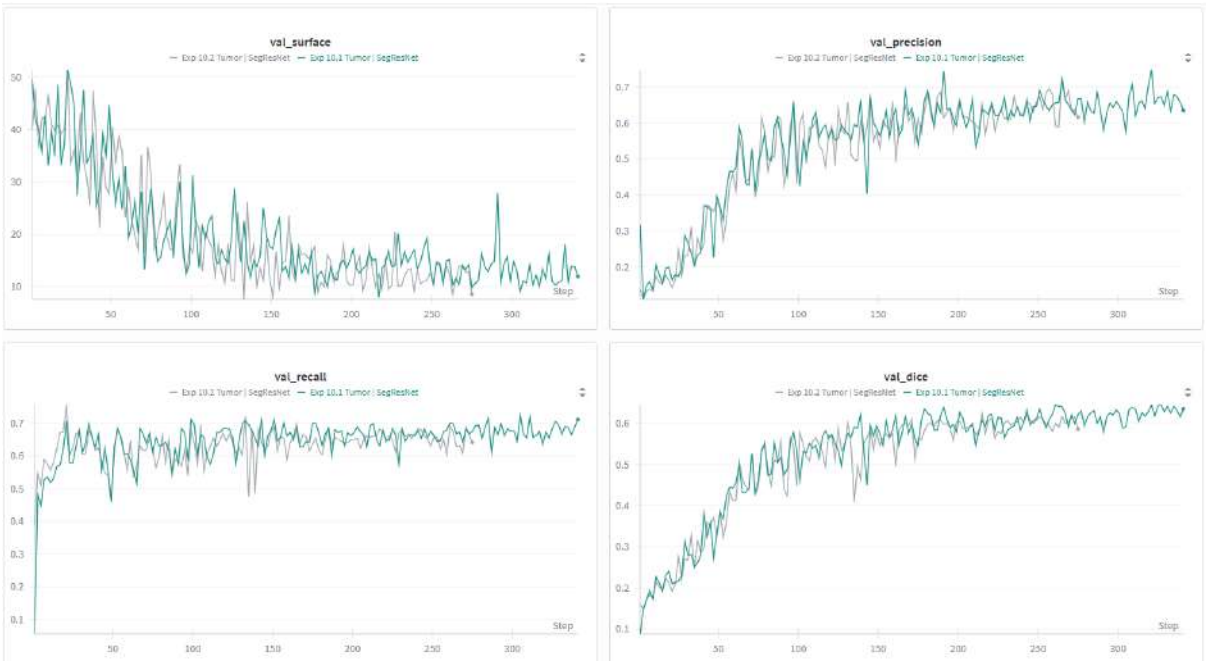

Figure 7.7: Loss curves

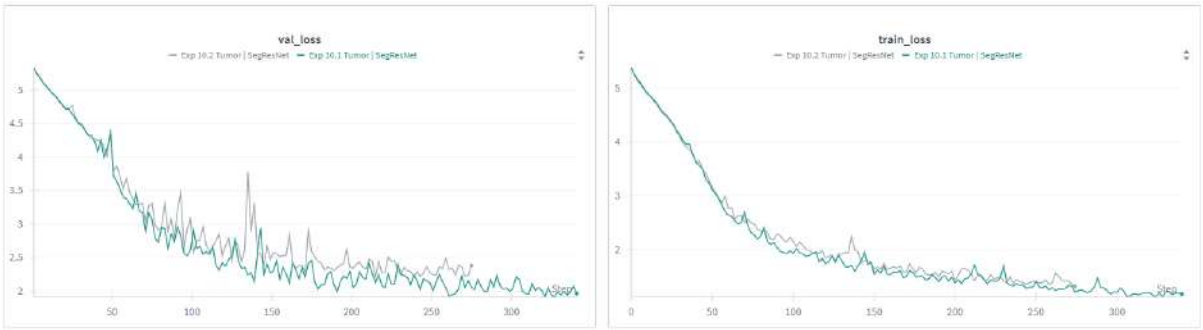

Figure 7.8: Metric curves

Fine-Tuning comparisons

As mentioned before, a SegResNet model was fine-tuned based on a completed training run for liver segmentation. The comparative results between this approach and the original SegResNet run are presented in Table 7.8.

Table 7.8: Final results

Parameter	**Fine-Tuned**	**Original**
Test Dice	0.683	0.652
Test Precision	0.693	0.614
Test Recall	0.722	0.75
BEST VAL DICE	0.653	0.646
Test Global Dice	0.745	0.633

The corresponding metric curves are shown below 7.9:

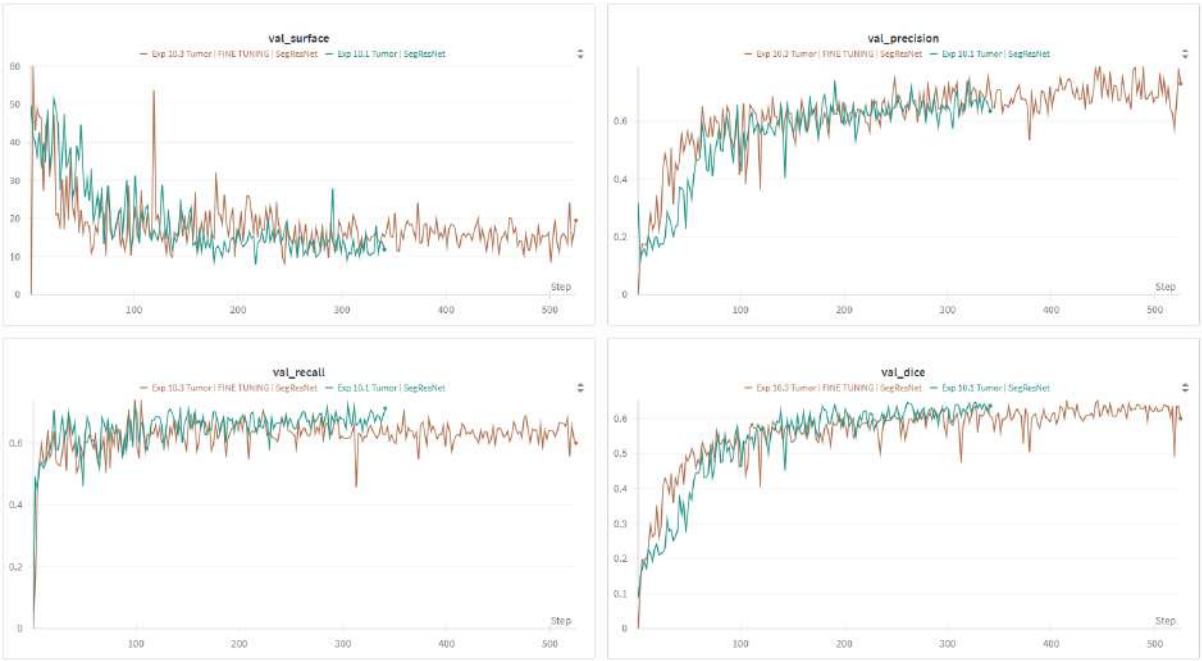

Figure 7.9: Loss curves

Another fine-tuning experiment was conducted, this time importing a model trained for 3D spleen segmentation from MONAI ZOO. Figures 7.10 and 7.11 present the metric results of the fine-tuned UNet architecture compared to the optimized UNet run executed on the remote computer. The head of the model should generally be re-initialized before beginning the fine-tuning, in order for the final layer to learn the brand new tumor features instead of the spleen features. The test metrics are shown in Table 7.9.

Table 7.9: Final results

Parameter	**Fine-Tuned ResUNet**	**ResUNet**
Test Dice	0.531	0.604
Test Precision	0.659	0.596
Test Recall	0.517	0.647
BEST VAL DICE	0.574	0.596
Test Global Dice	0.602	0.609

Figure 7.10: Comparison of metrics for fine-tuned UNet from MONAI ZOO

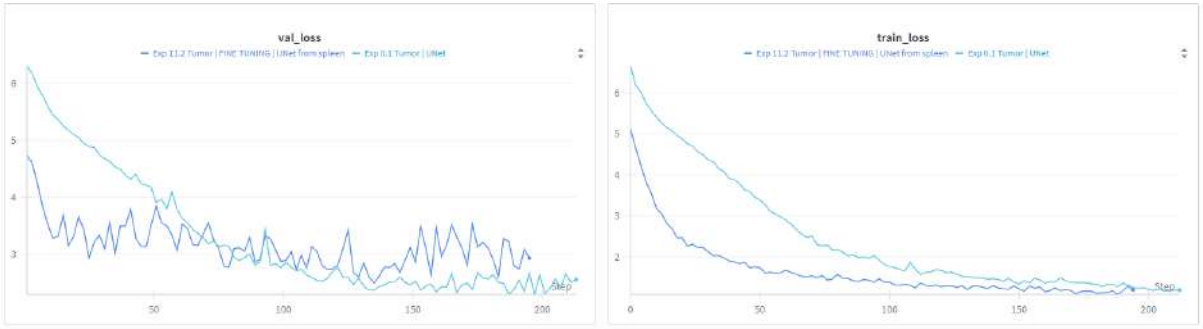

Figure 7.11: Comparison of loss curves for fine-tuned UNet from MONAI ZOO

7.2.3 Evaluation on test set

The best performing models were reloaded and evaluated on the test set, which originally contained 41 patients, but after optimization it had 22 patients. The model had never seen the specified patients during training. The best model is considered the regular SegResNet model, not the fine-tuned one for the testing phase.

Full vs optimized dataset

To highlight the differences in model performance between the complete and the optimized dataset, that only contains patients with more than 10 slices with tumor existence, we evaluated our best SegResNet model and plotted the DSC distribution across all patients to examine potential outliers with much poorer results than the average, in Figure 7.12. We can see that indeed the optimized dataset removes most of the potential outliers, along with some better candidates.

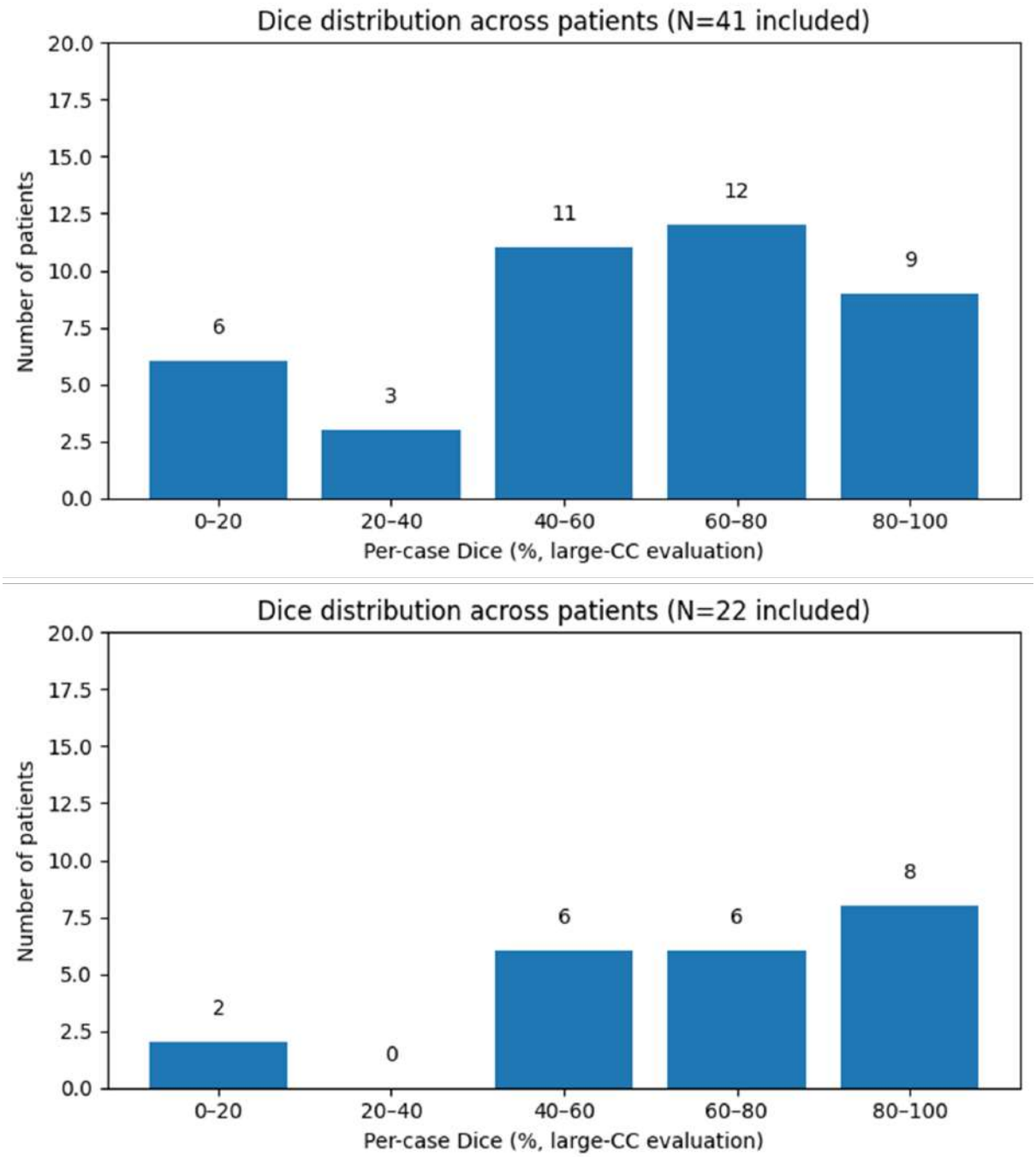

Figure 7.12: Outlier evaluation

Comparison among architectures

The following figures show the prediction masks by different architectures compared for the same patients.

Figure 7.13 shows the corresponding dice distribution for the additional architectures tested. SwinUNETR was not included in the computations due to its poor performance. The results verify that SegResNet had the least outliers.

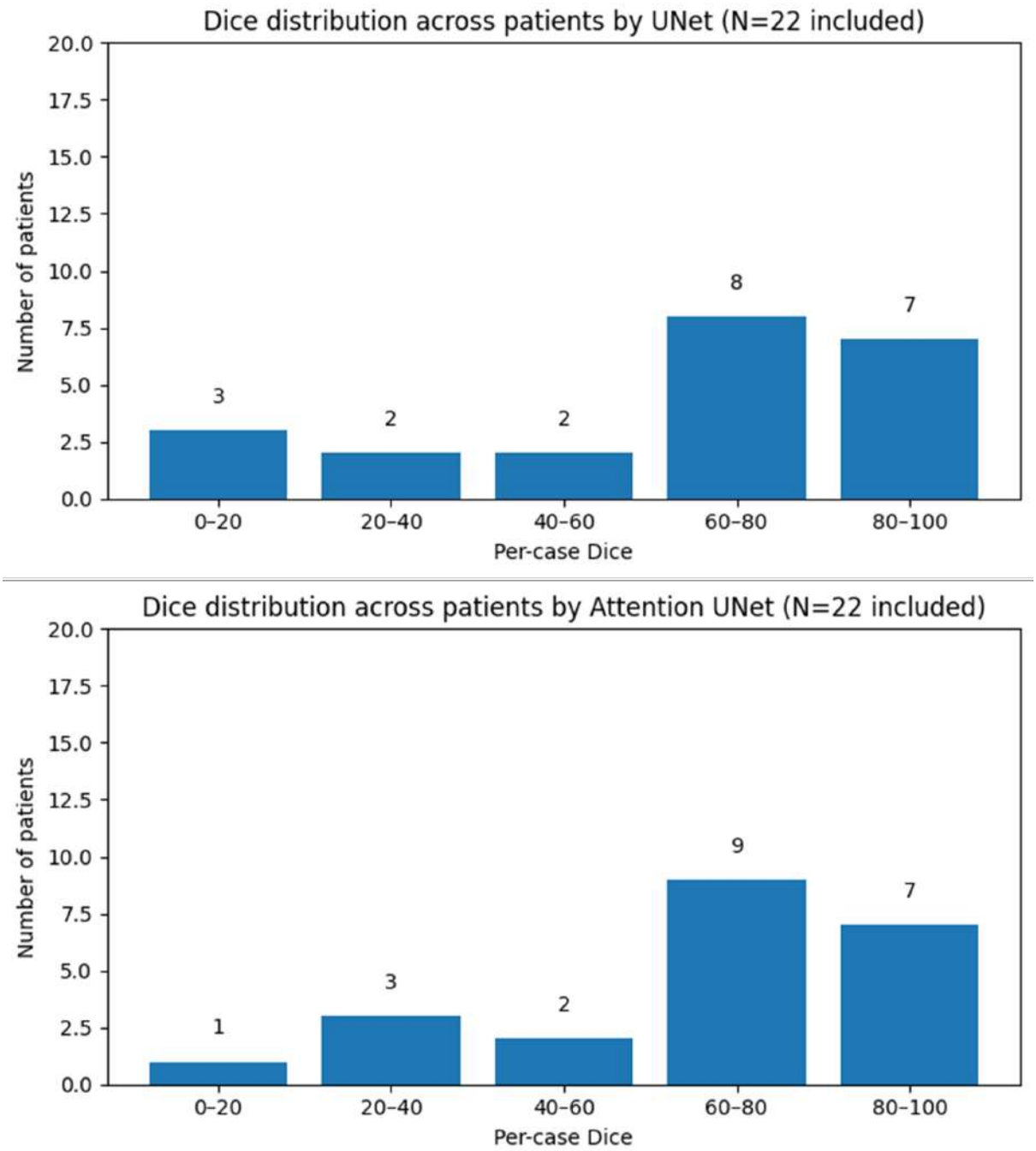

Figure 7.13: DSC bin among different architectures

Figure 7.14 shows an example where all architectures provided almost identical results of really accurate segmentations.

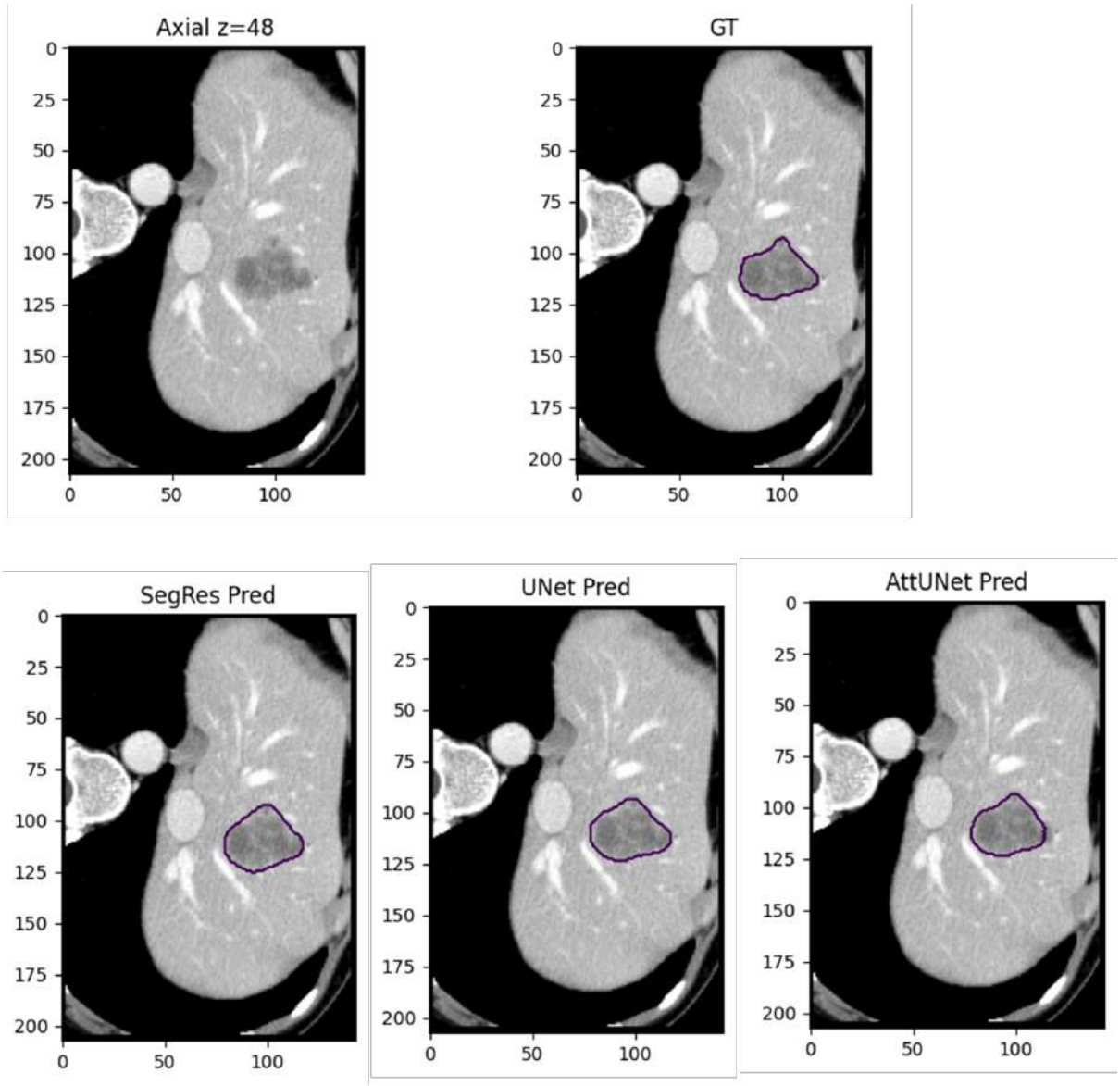

Figure 7.14: Successful segmentation by every architecture

Figure 7.15 shows an instance where our best model accurately predicted nothing, while the U-NET and Attention U-Net displayed FP instances, showing some over-segmentation tendencies.

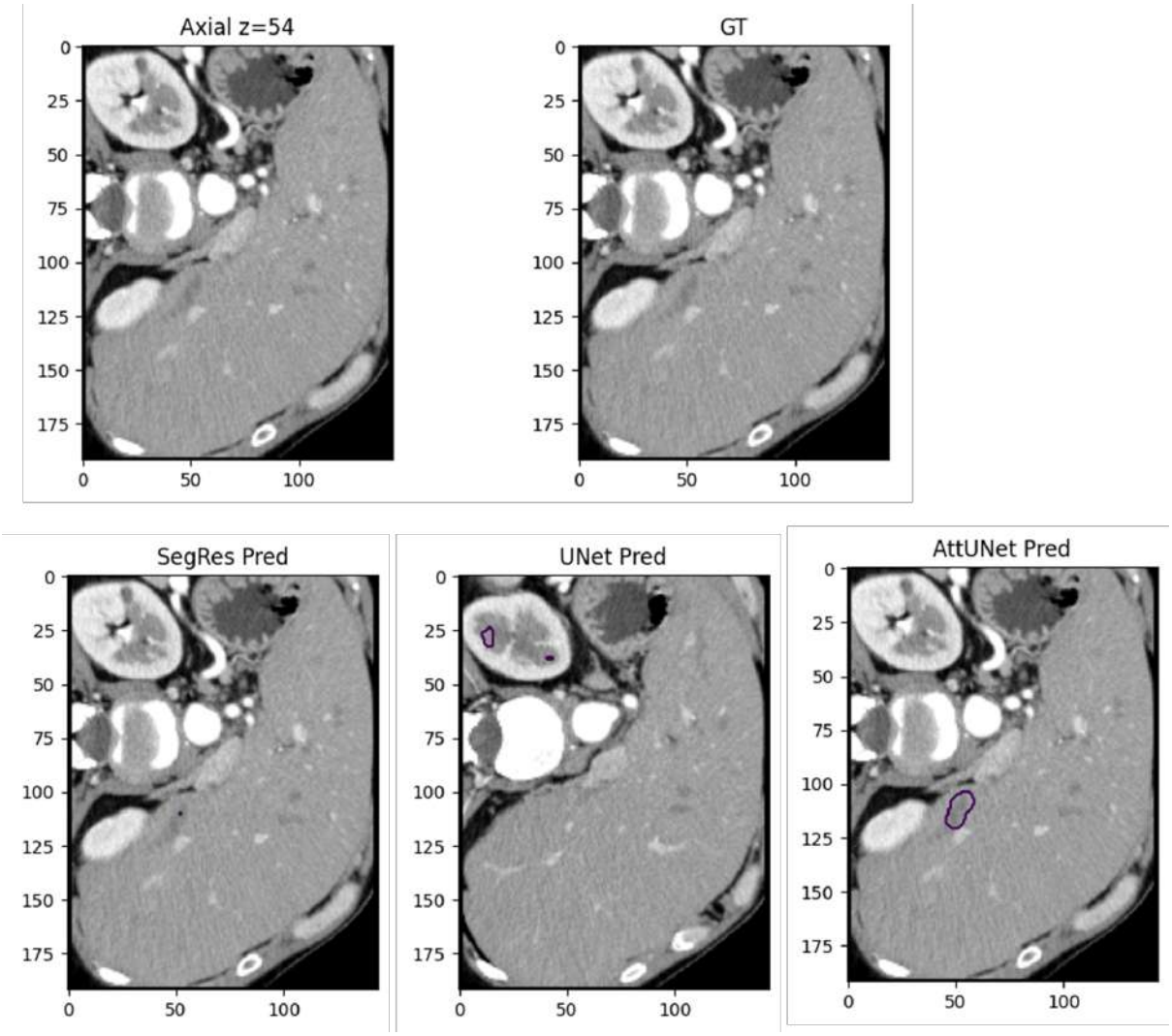

Figure 7.15: U-Net architectures over-segmentation example

Best architecture's results

The SegResNet architecture was the best one among all the trials conducted, displaying a final test DSC for tumor segmentation of 0.6527. That is why it was selected for thorough evaluation on the testing set. The metrics obtained during testing are presented in Table 7.10. We can observe a great variance between the min and max values, which pinpoints the variance in the shapes and sizes of the CRLM tumors, increasing the difficulty of this task.

Table 7.10: Testing results for SegResNet on optimized dataset (22 test volumes)

Metric	**Mean**	**Std**	**Min**	**Max**
Dice	0.6527	± 0.2220	0.0975	0.9041
ASSD	8.1884	± 10.9624	0.7891	49.0112
Recall	0.7514	± 0.2280	0.2037	0.9904
Precision	0.6143	± 0.2395	0.0641	0.9135
Global Dice		0.6329		

7.3 Final combined pipeline

7.3.1 Liver model tests

The best liver model, the SegResNet, was used to make predictions for the entire dataset. The predictions were then integrated into the complete tumor segmentation pipeline. Figure 7.16 shows the overlap between the GT mask (blue) and the predicted one (red), which is remarkably accurate.

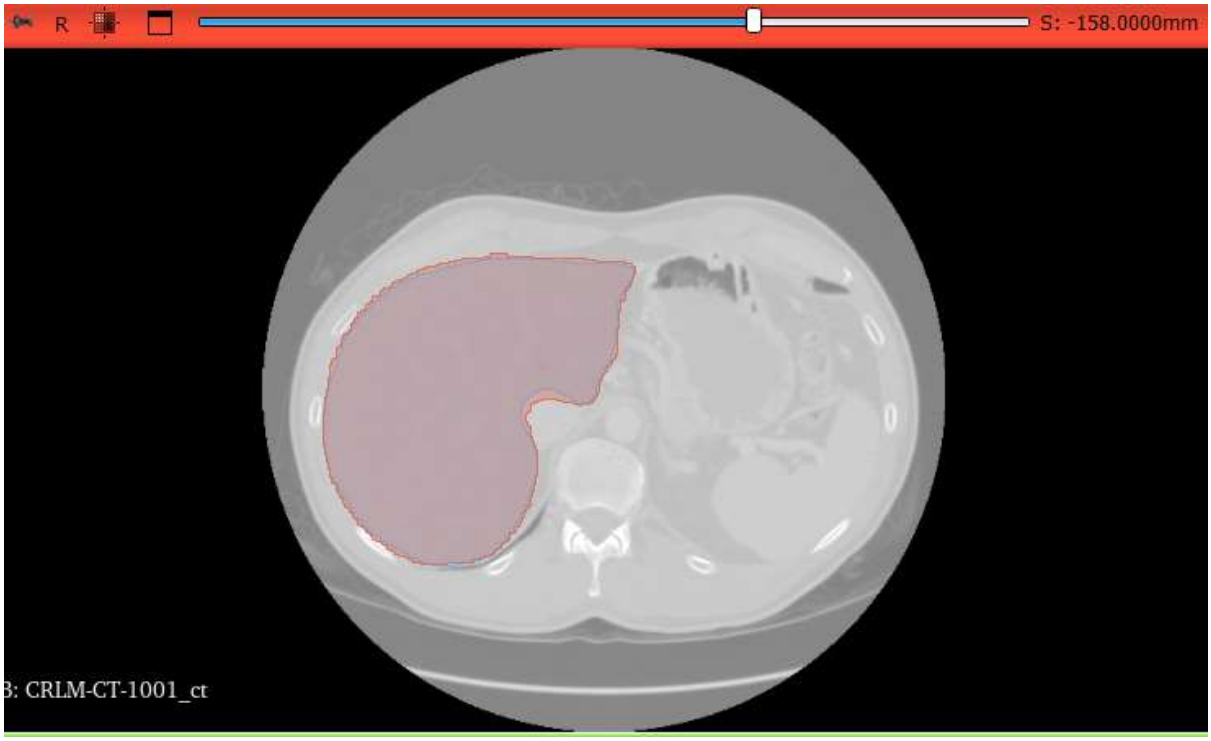

Figure 7.16: Visualization of predicted and GT liver masks

Table 7.11, contains the final testing results for liver segmentation, on all 197 patients of the dataset. These results verify that the predicted segmentations used for the next step of the pipeline are extremely close to the original masks. However, these results are not to be used for model evaluation and comparison, as the testing set contained patients that were part of the training set for the same model, which can result in slight inflation of the final results.

Table 7.11: Liver test metrics on entire dataset

Metric	**Mean**	**Std**	**Min**	**Max**
Dice	0.9682	± 0.0097	0.9081	0.9818
Recall	0.9687	± 0.0170	0.8994	0.9954
Precision	0.9681	± 0.0181	0.8522	0.9938
Global Dice		0.9690		
Global Recall		0.9688		
Global Precision		0.9692		

7.3.2 Final pipeline results

The final automated pipeline of two SegResNet models for liver and tumor segmentation can be visualized in Figure 7.17. This final automated pipeline illustrates a two-stage segmentation process in which the liver is first segmented from the input volume, followed by tumor segmentation within the localized liver region. This hierarchical design improves the accuracy of tumor detection by constraining the search space inside the liver.

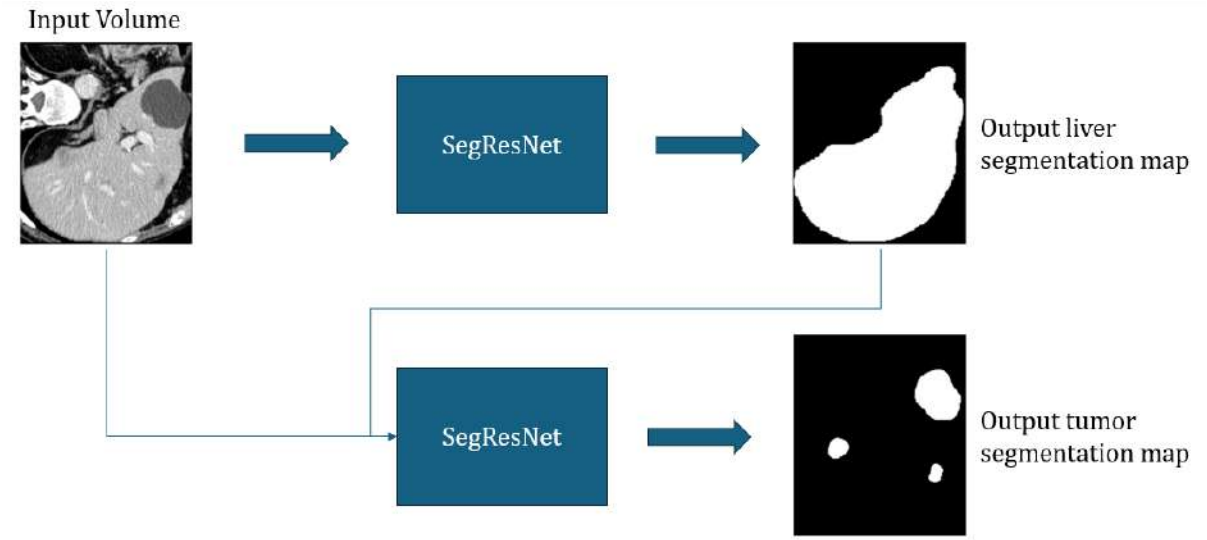

Figure 7.17: Diagram of final automated pipeline

The automated pipeline performed greatly on tumor segmentation, with a final test DSC of 0.6744. The metrics obtained during testing are presented in Table 7.10. We can observe a similar variance between the min and max values to the single channel SegResNet for tumor segmentation that was presented previously, however the final results are slightly increased for the final pipeline created.

Table 7.12: Testing results for final pipeline on optimized dataset (22 test volumes)

Metric	**Mean**	**Std**	**Min**	**Max**
Dice	0.6744	± 0.2092	0.0769	0.9271
ASSD	8.6173	± 9.6463	0.8004	37.2691
Recall	0.7604	± 0.1983	0.1841	0.9693
Precision	0.6473	± 0.2527	0.0486	0.9391
Global Dice		0.6631		

The DSC distribution for the final pipeline is presented in Figure 7.18. We can observed that the results are higher and more balanced than the original single-channel SegResNet trial.

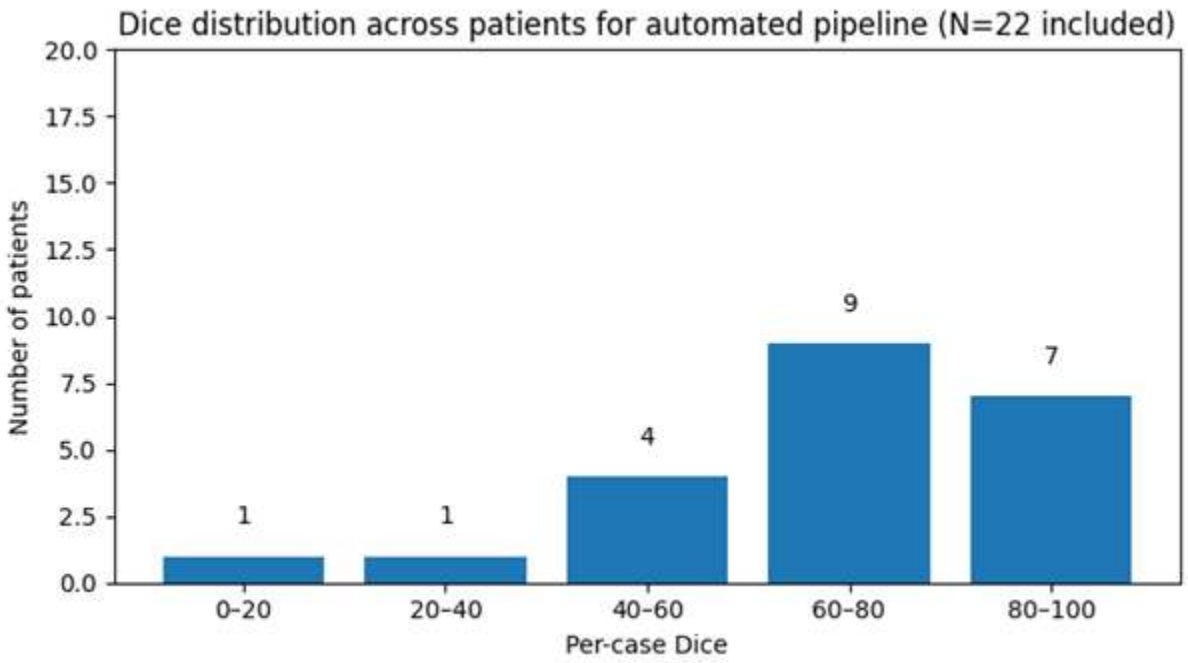

Figure 7.18: Test set DSC distribution for final pipeline

Lastly, we calculated the mean per-case DSC of this model, when excluding the two outliers with DSC lower than 40%. The result was a DSC of **0.7279** for 20 out of the 22 test set patients.

7.3.3 Final pipeline visualizations

The final model was also used to obtain valuable visualizations that help us observe the accuracy of the predicted regions. Some valuable examples are presented below.

Example 1 Figures 7.19, 7.21 and 7.20 contain some slices from the 2 outliers during the evaluation, which are prime examples of wrongfully segmented tumors. We can see that the model struggles to detect the tumors, mainly due to image contrast issues and label inconsistencies, resulting in false positive and negative predictions.

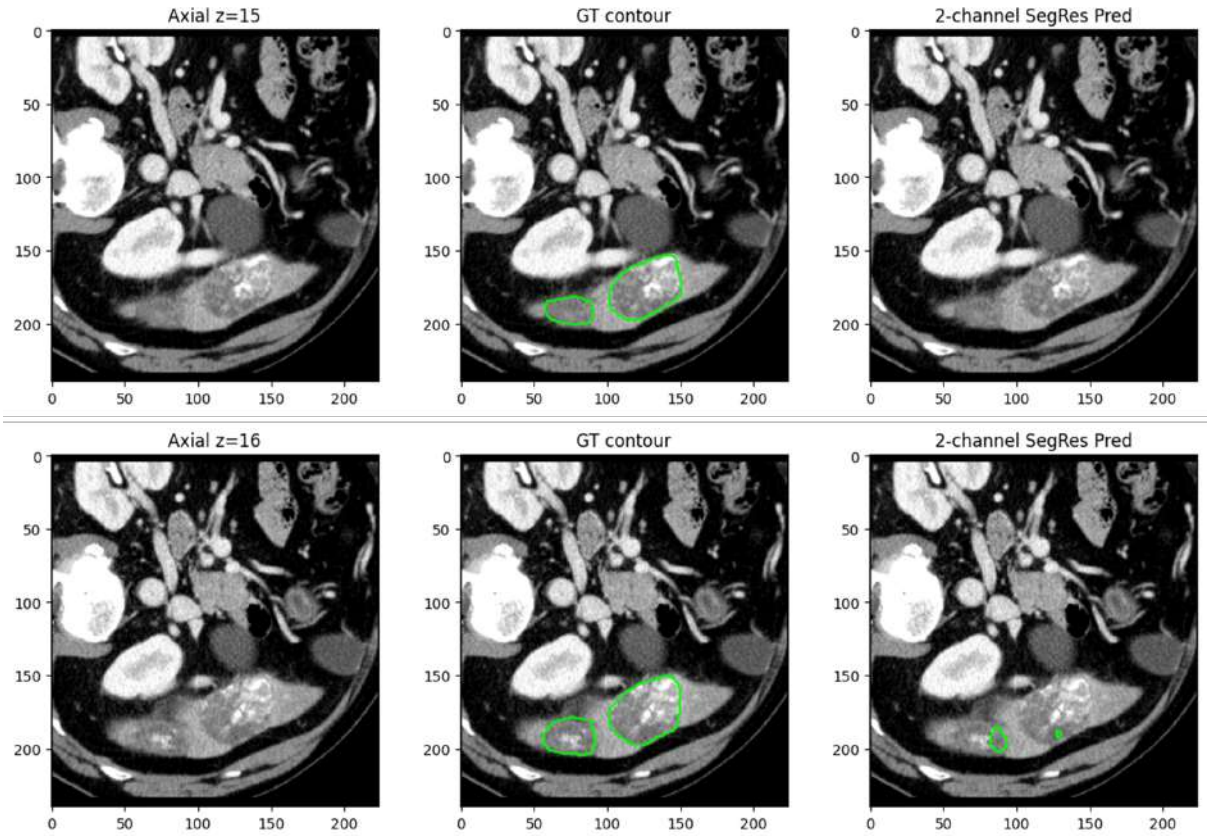

Figure 7.19: Example of difficult to segment tumors

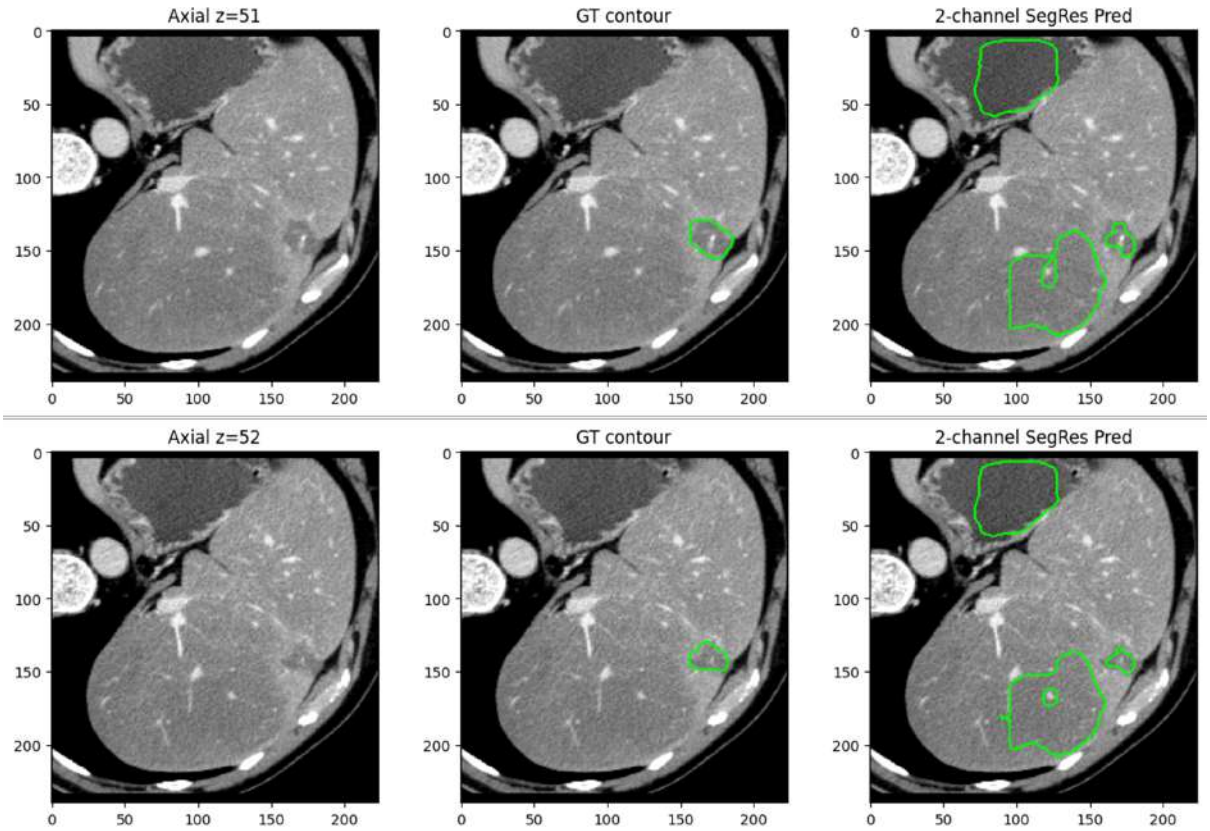

Figure 7.20: Example of difficult to segment tumors

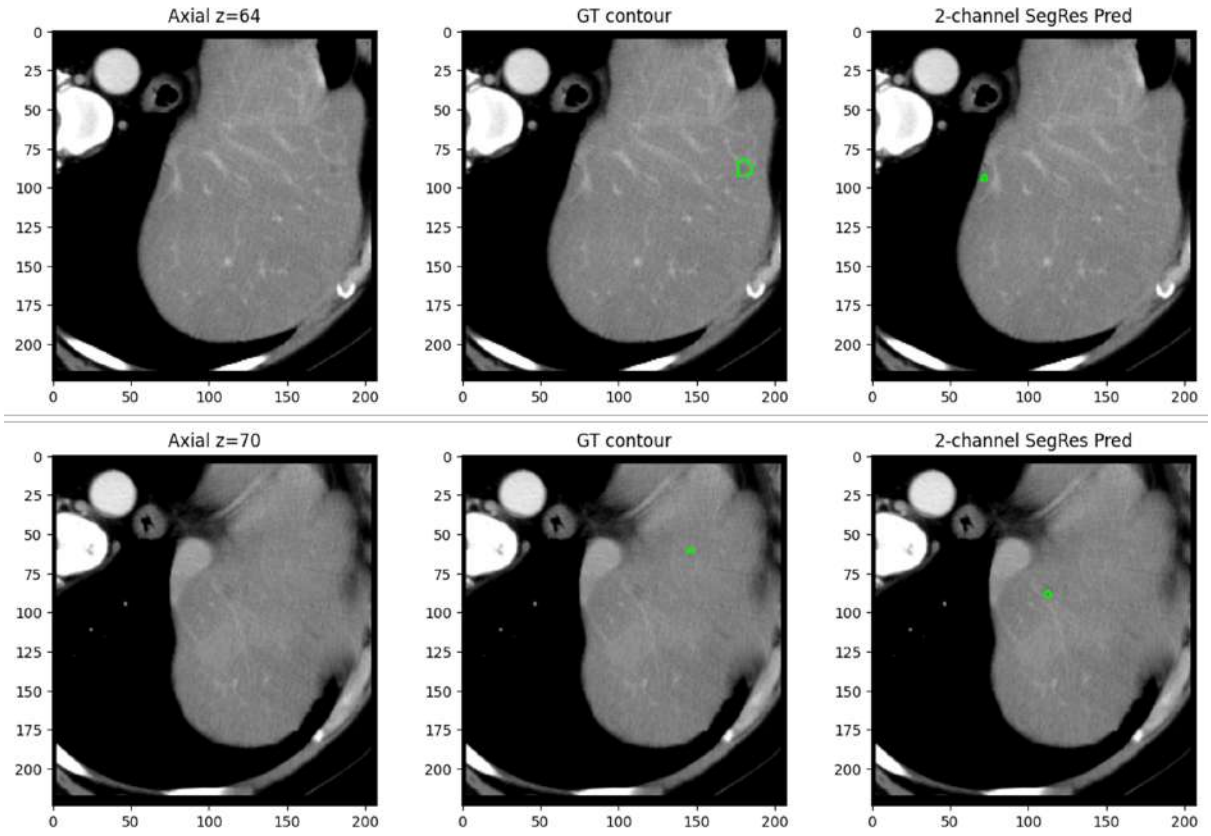

Figure 7.21: Example of difficult to segment tumors

Example 2 In the case of Figures 7.22, and 7.23, we have some slices of patients that belong to the highest distribution bin. The tumors are much larger and the scan's intensity makes them easier to detect. Even for smaller tumors, like in Figure 7.24, the model is still able to accurately segment the tumors for clear and consistent instances.

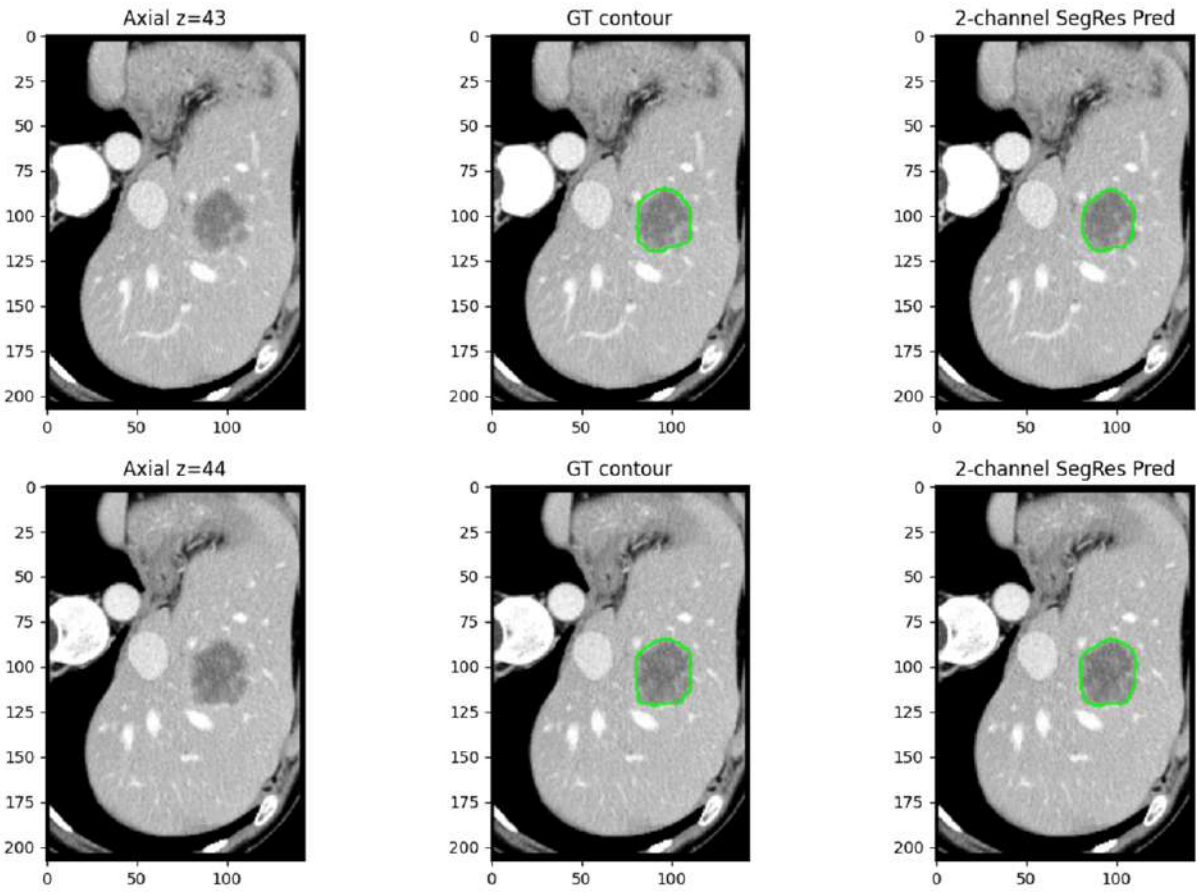

Figure 7.22: Example of accurately segmented bigger tumor instances

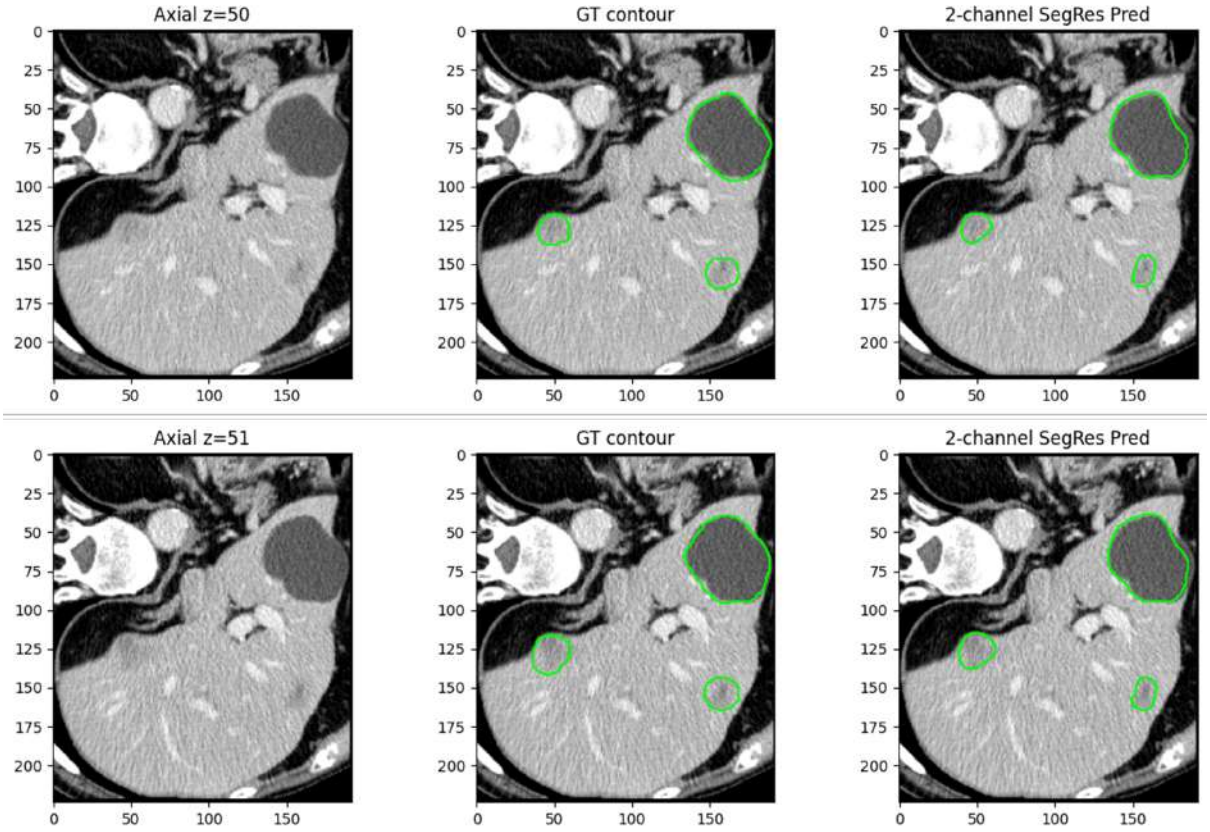

Figure 7.23: Example of accurately segmented tumor instances with different intensities

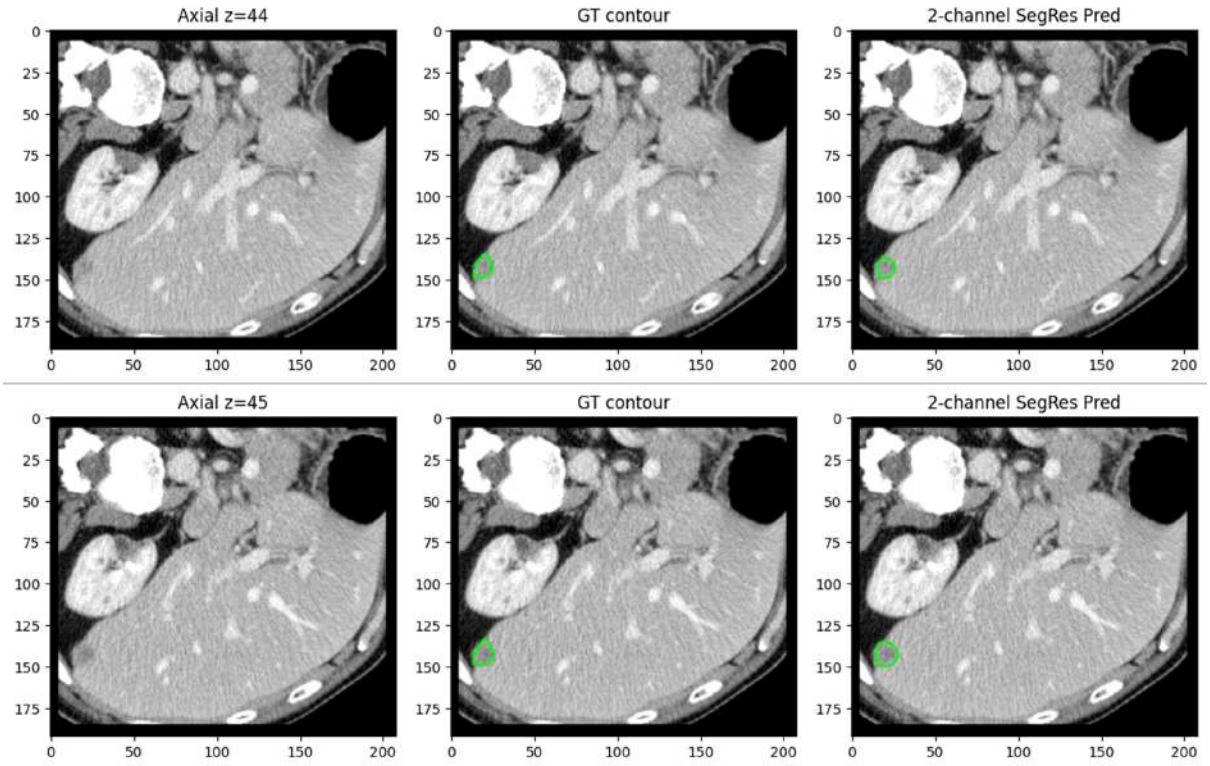

Figure 7.24: Example of accurately segmented smaller tumor instances

Example 3 Figures 7.25 and 7.26 show that even for patients with good dice scores, some ground truth labels can be misleading, as single tumor labels might be divided in half, or potential smaller tumor labels might be missing. Figure 7.27 showcases an extreme example where there is a label totally outside of the liver region, harming the final results.

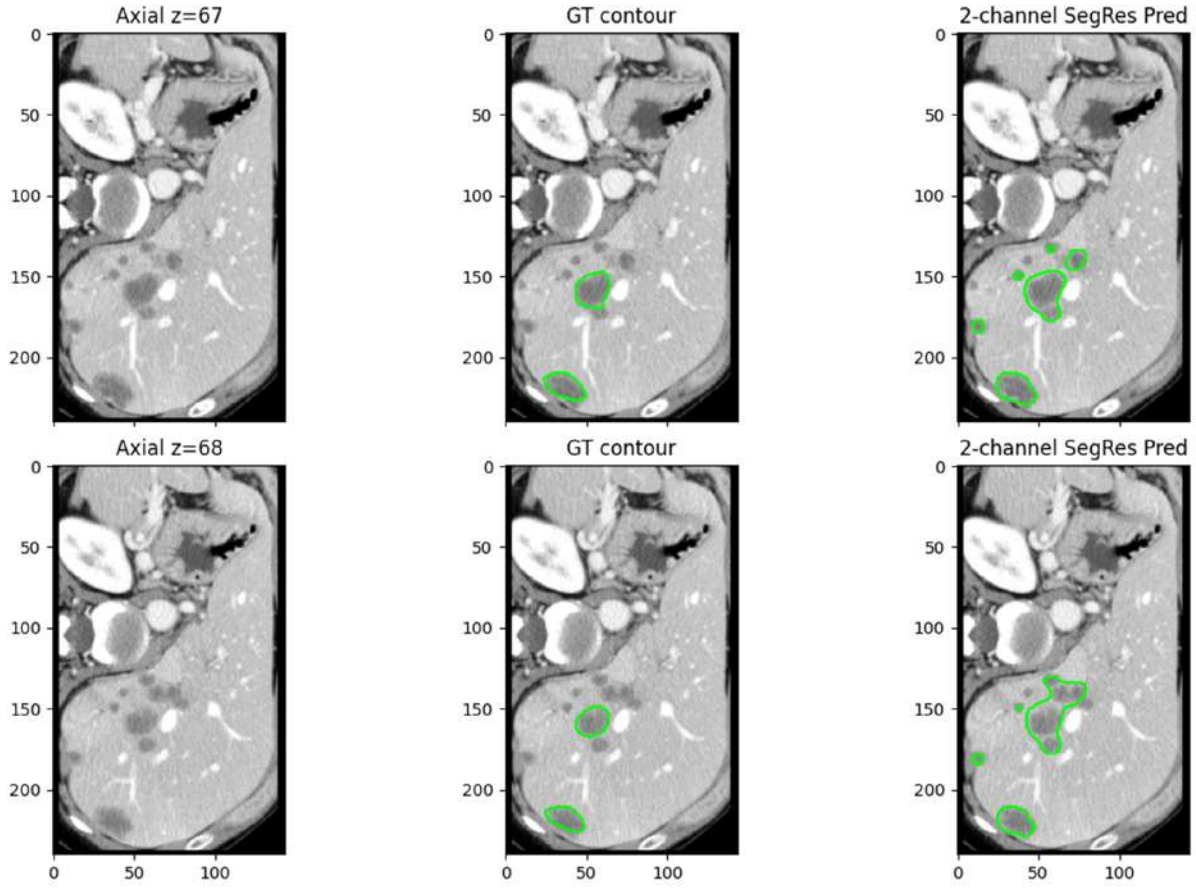

Figure 7.25: Example of multiple smaller tumor instances creating confusion

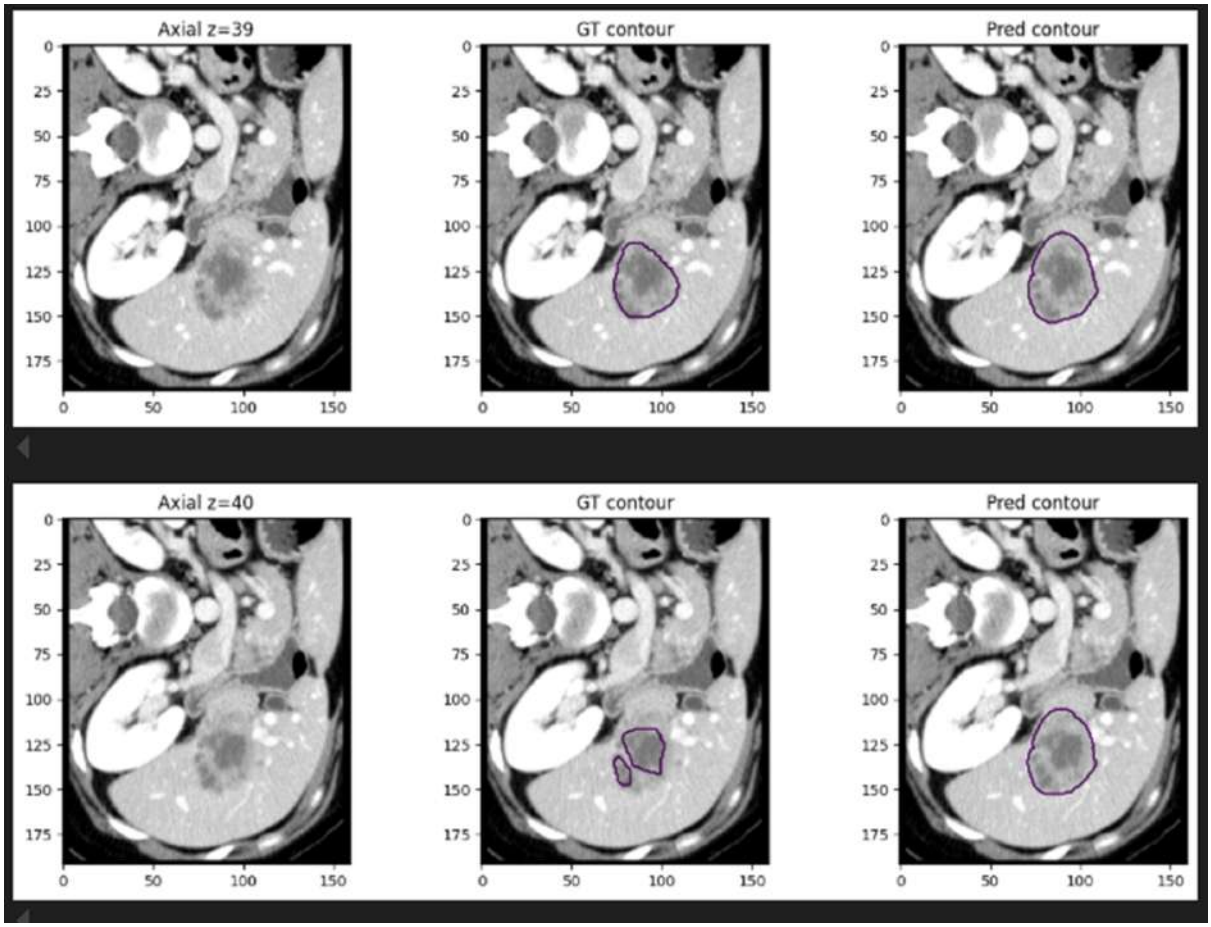

Figure 7.26: Example of sudden GT division

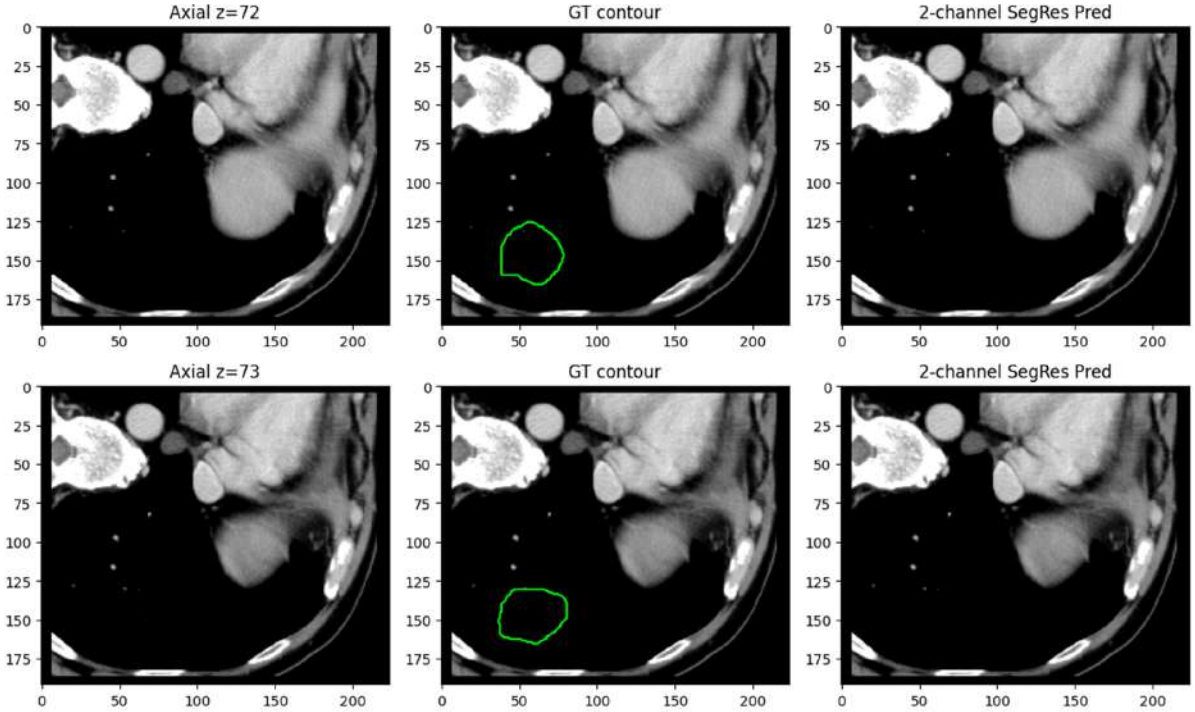

Figure 7.27: Completely missed tumor label

Chapter 8

Discussion

8.1 Interpretation of results

8.1.1 Liver segmentation

The liver segmentation results are consistent with state-of-the-art standards and can be compared to the reported results from other relevant studies in Chapter 5. The validation graphs across different architectures are quite similar,

All tested architectures performed effectively and consistently in the liver segmentation experiments. Models converged quickly during training, as shown by the validation curves, and validation Dice scores plateaued around high values after a relatively small number of epochs. The preprocessing and augmentative techniques were successful in stabilizing training, as displayed by the validation curves, which showed steady improvement without overfitting. SegResNet was chosen for the final evaluation on test data because it performed slightly better than the other models.

The SegResNet model performed greatly on the unseen test data consisting of 41 patients, with a mean Dice score of 0.944 and very low variance across cases (standard deviation of 0.029). The low mean average symmetric surface distance (ASSD) of 1.01 mm verifies the accuracy of the volume boundaries predicted. The network's precision (0.967) and recall (0.888) further demonstrated its sensitivity in detecting liver voxels and accuracy in preventing false positives. The model was consistent across patients, with few outliers, as evidenced by the narrow range between minimum and maximum values.

The validation curves displayed almost similar results across all architectures. This can be explained by the nature of the liver segmentation task, as the liver is a large, well-defined organ that is uniform in shape and appearance across different patients. The liver region is clearly displayed in many slices across all patients, therefore the specialization of each architecture does not have a great effect on the final performance. All encoder–decoder structures are able to learn the strong spatial and intensity characteristics that differentiate the liver from surrounding tissue.

Overall, the performance of the liver segmentation model can be considered state-of-the-art and competitive with values reported by related literature. The network was able to accurately segment the liver region even in cases of anatomical or intensity differences between scans. These results are particularly important because they provided a reliable foundation for the subsequent tumor segmentation experiments.

8.1.2 Tumor segmentation

The experiments around tumor segmentation were more thorough and specialized, as it is a much harder task than liver segmentation. The decision to exclude the background seems to have hurt the results, however makes them more representative of the actual capabilities of the model on tumor segmentation without inflation from background segmentation success. This is an important aspect, as some relevant studies do include the background in the results and therefore can lead to overestimation of their model's ability.

Another important result, is the *comparison between the optimized and regular dataset*. Excluding 87 out of the 197 patients that had fewer than 10 tumor-positive slices, gave us the chance to over-sample patients with more tumor instances, in an effort to deal with the background domination. The training and validation curves provide a valuable demonstration of the results of this method. The training curves (loss and Dice) for the optimized dataset (purple) show smoother and more stable convergence compared to the original method (brown). Training loss decreases steadily, while training Dice rises higher, suggesting that the model learns more consistently when the dataset contains fewer extremely sparse cases. Similarly, validation Dice starts higher, increases quicker, and maintains a more stable plateau compared to the original dataset experiment. This indicates that removing patients with almost no tumor voxels reduces the severe class imbalance that would otherwise bias the model toward background prediction. Similarly, validation loss declines faster and to lower values, verifying that the optimized dataset improves generalization.

The availability of the remote PC with an NVIDIA RTX 4080 16GB GPU was necessary for a complete experimentation process. The enhanced GPU of the remote machine, which allows for a greater spatial size during training. Compared to the local GTX 1650 4GB, the larger memory capacity of the RTX enabled the use of larger spatial sizes and number of samples per patient (up to $160 \times 160 \times 64$ and 12 samples for UNet). This enabled the models to capture more volumetric context in each epoch, improving both efficiency and segmentation quality. The training logs highlight the immense difference in runtime. While a single epoch run on the GTX for a SegResNet model could take around 20-25 minutes, the RTX reduced this time to less than 1 minute. This speed-up was crucial, as it allowed longer training schedules (200-300 epochs) and experimentation with more complex architectures such as SegResNet and SwinUNETR.

The *architectural experiments*, with four different implementations, also provided valuable insights. Four different architectures were tested with similar configurations and hyperparameters like loss, activation function, learning rate, maximum epochs, patience, sliding window overlap and evaluation metric. However the spatial configurations of the inputs during preprocessing were different depending on the memory constraints for each architecture. The testing results and training metric curves obtained, provide valuable insights on each architecture's performance.

The SwinUNETR model, struggled to follow the performance of the other metrics and was the first to trigger the early stopping mechanism. It achieved the lowest scores across dice, recall, precision and surface distance, by a margin. This can be attributed to the large memory requirements and sensitivity to limited data that characterizes heavy transformer-based models, like the implemented one. Due to the limited hardware resources. even for the remote PC, we had to significantly lower spatial size and number of samples, which seems to have hurt the final performance.

ResUNet and Attention U-Net performed moderately well, with Dice scores around

0.60–0.64. The curves show that both models converged smoothly and without many fluctuations. Attention U-Net, in particular, demonstrated the highest precision (0.665) and global dice, but lower recall and per-case dice which indicates that it was successful at predicting large and certain tumor regions, without many false predictions, but it missed some lesions.

Lastly, SegResNet was the best model in this experiment, achieving the highest test Dice (0.652), recall (0.75), and validation Dice (0.663). The validation curves for SegResNet were quite stable, with both Dice and loss trends showing steady convergence. Its high recall suggests that SegResNet was better at detecting small and heterogeneous tumor regions, a critical factor in CRLM segmentation where under-segmentation can result in clinical implications. Precision remained at a satisfactory level above 60%, along with the final average surface distance metrics, around 10-20 mm, confirming the accuracy of the final boundaries.

Overall, the results confirm that CNN-based residual networks maintain a strong balance between capacity, stability, and computational cost, which makes them better more appropriate for tumor segmentation under limited resources compared to more computationally demanding transformer models.

The *loss comparison* conducted for the best model architecture (SegResNet) between Dice-Cross-Entropy and Dice-Focal losses provided additional insights on the model behavior. The training curves highlight the stable convergence achieved in both experiments. However, the quantitative results for DiceFocal loss were slightly better for final test Dice (0.652 vs. 0.63), precision (0.614 vs. 0.582), and global Dice (0.633 vs. 0.604), while maintaining recall at a similar level (0.75 vs. 0.761). These findings suggest that the additional weighting mechanism introduced by the focal component helps the model to better handle the class imbalance of this task. Even though the final results are similar, DiceFocal displays remarkable consistency across metrics, making it the more fitting choice for our task.

Evaluation on test set

The distribution comparison between the original and optimized datasets verifies that some of the patients in the original dataset were not suitable for this segmentation task, as the small size or frequency of tumors on their scans drove the final DSC below 40%. In contrast, the optimized method has successfully excluded most outliers, with only 2 patients performing under 40%. Also some average-performing patients were excluded, but almost all patients from the top bin remained in the optimized dataset.

The distribution of dice values among SegResNet, AttentionUNet and ResUNet architectures, indicates that SegResNet is better at avoiding outliers and achieving overlap over 40% between GT and predicted masks, while the other two architectures display a smoother dice distribution across all bins. The visualized slices suggest that all architectures are capable of accurately segmenting large, well-defined tumors with obvious intensity and contrast differences from liver tissue. However, especially U-Net architectures, seem to tend to over-segment on cases with poorer contrast, resulting in increased FP.

Fine-tuning

The first fine-tuning experiment, which utilized weights from a pretrained SegResNet model to adapt the architecture for tumor segmentation, demonstrated a performance

improvement compared to training from scratch Fine-tuning raised the test Dice from 0.652 to 0.683 and the global Dice from 0.633 to 0.745, while also boosting precision from 0.614 to 0.693. Recall was slightly lower, yet still competitive. These results highlight the effectiveness of transfer learning across related segmentation tasks. The dice curves show that the fine-tuned model was able to grasp tumor-specific features faster and reach a smooth plateau. However, fine-tuning from the same dataset, can come with some risks, as the network may reuse specific dataset features from pretraining, resulting in inflated performance on the test set but limited generalization to external cohorts.

The second fine-tuning experiment showcases the limitations of transferring knowledge from spleen segmentation to CRLM segmentation. While the fine-tuned UNet from MONAI Zoo achieved slightly better precision from the baseline ResUNet, it displayed much lower recall and overall Dice (0.531 vs. 0.604). This suggests that the model struggled to adapt to the heterogeneity and smaller size of CRLM lesions, showing that the features learned from spleen data do not generalize well to tumor structures. The training and validation curves also show higher instability and noisier convergence, as even if dice starts from higher values it does not improve substantially throughout the training process.

In conclusion, the fine-tuning results indicate that the selection of the appropriate dataset for pretraining is quite important for the success of the transfer-learning process. An effective balance must be found between using data that is closely related to the target task and without harming the final generalization abilities of the model.

8.1.3 Final pipeline results

The final pipeline created for automated tumor segmentation, with two input channels, performed even better than the single-channel trials. The final test set per-case DSC of **0.674**, is slightly lower, yet comparable state-of-the-art models for CRLM segmentation, like [100, 105]. The final recall (0.759) and precision (0.647) values are also improved, suggesting a balanced model, focused on avoiding FN results that can be quite serious for medical imaging tasks, as they represent missed tumor instances.

The improved results, can be partially attributed to the second input channel added, with the liver predictions. The final model combines the spatial features from the predicted liver masks that indicate the location of the tumors, with the more specific characteristics from the CT volumes that contain important shape and intensity information. Additionally, leakage of information from GT masks is reduced, as the liver predictions are used to spatially guide the model.

The examples visualized for our best model help better understand the dataset and the specific characteristics that harm the model's performance. The model seems to be oversensitive to small intensity variations, possibly mistaking normal parenchymal texture, vessels, or imaging noise for lesions. Additionally, it seems to miss lesions with low contrast to background, regardless of their size. Lastly, sometimes the model manages to capture the size and shape of smaller lesions but misplaces them inside the liver region.

In contrast, the model successfully predicts the shape, location, and extent for one or multiple tumor instances when there is adequate contrast from the surrounding tissue. There is a small tendency to under-segment some lesions, which however does not result in increased FN.

There are also some examples of poor GT labels that prevent the model from increasing its accuracy. In some cases the ground truth contouring appears inconsistent. For

example, a single tumor label is split into two disconnected contours in the next slice, even though the intensity pattern suggests continuity. In another example, the ground truth labels appear suspicious because they are drawn far outside the liver parenchyma, in regions without any visible lesion, resulting in false negative predictions by the model, which does not detect any tumors. These annotation quality issues can mislead model training and evaluation.

8.2 Main limitations observed

During the experimentation phase of the 3D CRLM segmentation task, we faced several limitations and practical challenges. A major constraint was the long training time caused by the computational demands of 3D volumetric data. Training required substantial memory and compute power, but the local resources were limited to a 4 GB GTX GeForce GPU, which restricted batch size and model capacity. Close to the end of the project remote access to a more powerful 16 GB NVIDIA GEFORCE RTX GPU was granted, allowing the experimentation with heavier models and parameters, however the availability of resources was still limited.

Another important observation was the class imbalance in the dataset. Many patients had only a handful of slices containing tumor voxels, and some tumors were extremely small compared to the liver volume. This imbalance caused the network to struggle with sensitivity, with the average values around 60%, since the model struggled to detect and successfully segment sparse tumor instances. In addition, there was a great variation in scan depth, intensity ranges, and liver shapes, which complicated preprocessing and normalization.

Ground truth labels also presented some inconsistencies, as in several cases, tumors were extremely small, or they appeared and disappeared inconsistently across consecutive slices, making learning even harder. Other cases included clearly, mislabeled tumor instances and poor quality scans with bad contrast between tumors and surrounding tissue or other anatomical features, that overall confused the model and resulted in increased FP and FN predictions.

The experimentation process of different modules and methods was quite resource intensive. Many architectures and hyperparameters needed to be explored, the limited GPU availability made it hard to conduct thorough trials of various different methods. Additionally, many training strategies required great modifications to the base training code, for example handling inputs as one-hot encodings or adjusting the training loop, which added extra complexity.

Overall, these constraints shaped the experimental strategy. Certain lighter architectures, like U-Net had to be prioritized and utilized for most experiments, while the focus was given on delivering accurate and objective results, with minimum editing of the original dataset or leveraging of the dominant background class, which boosted results.

8.3 Comparison with literature

When comparing our best results with existing literature, as described in Chapter 5, a clear distinction emerges between liver and tumor segmentation performance.

For liver segmentation, our pipeline achieved a Dice score of 0.968 with high precision and recall (0.969), which is comparable to some of the best-performing reports. This

confirms the success of our model in the relatively easier liver segmentation task.

In contrast, tumor segmentation was substantially more challenging. Our final automated pipeline for liver and tumor segmentation, with two SegResNet models trained with DiceFocal, achieved a final tumor DSC of 0.674, precision of 0.647, and recall of 0.76 on the test set. These values are similar to the most subjective and rational CRLM experiments, like [105], who reported a Dice of 0.73 and precision of 0.44, [101] with DSC of 0.655 on 3DIRCADb and [100] with a DSC of 0.74 on larger tumor instances. The differences with other studies can be attributed to the smaller size and variability of our dataset, since many literature works rely on larger or more curated public datasets (e.g., 3DIRCADb, CAIRO5). Also the hardware constraints limited our abilities for extensive experimentation with more models and parameters. Lastly, some studies applied extensive dataset optimization techniques, removing non-tumor slices or patients [103], or even included the background on the metric calculations, inflating the final results [51, 85, 91].

Chapter 9

Conclusions and future Work

9.1 Summary of achievements

The main goal of this project was to develop a reliable and automated pipeline for the segmentation of CRLM in 3D CT scans. After thorough experimentation, a complete workflow was designed and implemented with the help of MONAI framework, accompanied by a literature review and theoretical study that explained and justified the experimental methods and strategies followed. The final test DSC of **0.674** is highly satisfactory, based on the implications of the segmentation of this specific type of cancer.

The liver-tumor segmentation task can be solved in an organized and effective manner by putting in place a two-stage pipeline. The tumor model functions within a clearly defined region of interest when the liver segmentation is used as a prior, improving accuracy and lowering false positives. In addition to increasing overall performance, this approach simplifies the inference process, which makes it more useful for clinical workflows where accuracy and dependability are crucial.

Implementing a two-stage pipeline provides a structured and efficient solution to the liver–tumor segmentation task. The liver segmentation is used to guide the tumor model's operation within a well-defined region of interest, which enhances accuracy and reduces false positives. This strategy also streamlines the inference process, making it more practical for clinical workflows where precision and reliability are essential.

Using 3D segmentation allows the model to leverage volumetric context and spatial consistency of the tumors within the liver. This is particularly important for CRLM, where lesions can be small, have variable shapes, and appear across multiple slices.

The literature review included the medical background of colorectal cancer and CRLM, the fundamentals of DL, the specifics of segmentation tasks in medical imaging, and an overview of the most commonly applied architectures in the field. This review helped to establish a solid foundation for the experimental phase, while also highlighting the clinical importance and challenges of CRLM segmentation.

Based on this, a full training pipeline was implemented. Special attention was given to preprocessing and augmentation, since the dataset presented issues such as class imbalance, varying slice depths, and heterogeneous tumor appearances. A range of transformations were applied to normalize intensities, standardize input dimensions, increase tumor positive samples and ensure robustness through data augmentation. The main evaluation metric used was DSC along with recall, precision, and surface distance, to ensure reliable performance evaluation.

Various different architectures like U-Net, UNETR, and SegResNet were tested under

multiple configurations. U-Net, being the lighter one of the models, was also used for experimentation with training strategies and hyperparameters. The model primarily relies on differences in intensity and texture within CT images to successfully segment the tumor regions, since tumors often appear with contrast or brightness variations relative to surrounding liver tissue. Through convolutional layers, it captures high-level spatial representations, which help distinguish tumors from neighboring vessels or imaging artifacts.

The most effective architectures were identified separately for liver segmentation and tumor segmentation, mainly based on validation and test set DSC results. Recall results were also important for model selection, since they reflect the model's ability to avoid leaving lesions undetected. These models were then combined into an automated two-stage pipeline, in which the liver model's predicted masks were used to guide the tumor segmentation task, as second input.

The results were presented in a systematic way with various graphs and tables deriving from Weights & Biases framework, to illustrate model performance across the different experiments, allowing for clear comparisons and objective conclusions. The evaluation process was objective and robust, as we avoided inflating results, by excluding the background class from the Dice metric and by applying a consistent dataset optimization method, rather than picking specific patients for exclusion.

In terms of performance, the liver segmentation model achieved state-of-the-art results, with Dice scores approaching 96%. Tumor segmentation proved much more challenging. It was especially affected by the strong class imbalance between the background and the relatively small and often sparse tumor lesions. In many patients, only a few slices contained visible tumors, while in others the tumors appeared as very small or fragmented structures, making them difficult to capture consistently. Additionally, the use of 3D volumetric data substantially increased training times and memory requirements. However, the final tumor results, with a DSC of 67,4%, are competitive and comparable to values reported in related literature. These results demonstrate that, despite limited resources and dataset constraints, the developed pipeline is capable of reaching a strong performance level.

Overall, the project effectively produced an automated 3D CRLM segmentation pipeline, supported by a carefully structured methodology, extensive experimental validation, and a thorough literature review. The results offer a useful application as well as a significant addition to the research of deep learning applications in medical imaging. Our research provides a strong foundation for CRLM-specific tumor segmentation, while also highlighting opportunities for improvement through larger and more representative datasets, more specialized pipelines, and enhanced hardware resources.

9.2 Future research directions

Even though this project displayed feasible methods for 3D liver and CRLM segmentation with limited resources, there are still a number of areas with potential for future research, from enhanced data and computational infrastructure, to advanced training methods, architecture implementations and clinical validation.

Future studies could benefit from more complete and diverse datasets, with an increased number of patients and tumor instances and fewer class imbalances. Smaller tumor instances could be combined and multi-center datasets could be implemented to

enhance performance. Additional validation and evaluation from experts could improve the segmentation quality of the datasets which could help the model with feature extraction. For example, the use of Total Tumor Volume (TTV) as a clinically meaningful biomarker, unlike the voxel overlap used in standard segmentation tasks, could enhance the compatibility of research segmentation tasks with prognosis and treatment monitoring applications [106]. Moreover, more specialized strategies for case selection among the provided data could be implemented, that are focused in the label accuracy, image quality and intensity specs of the original volumes.

Another area of promise is the construction of more specialized architectures and strategies for the specific CRLM task. The curriculum training approach proposed by [82], implements a U-Net variation with two branches, an over-complete one to evaluate small structures and an under-complete one for higher-level structures, while training starts from easier samples and then gradually expands to harder ones. This method can help the model understand all types of tumors in a more organized and accurate way. [109] proposed an enhanced U-Net++ architecture with ECA attention module for channel emphasis and deep supervision, which seems to improve segmentation of blurry boundaries and handle complex gradients effectively.

A promising direction for future work in 3D CRLM segmentation is the development of fully modular and automated training pipelines that reduce the need for manual code edits. [110] proposed a framework that gathers the full training, inference, and evaluation workflows by the user, without underlying code changes needed and provides support for advanced techniques like patch-based learning, test-time augmentation, and model ensembling. Similarly, [111] suggested the 'Auto-nnU-Net' architecture offering automated hyperparameter decisions and optimization strategies that balance accuracy against time complexity. [106] proposes a self-learning pipeline (teacher–student setup), where limited manual labels are leveraged to generate large pseudo-labeled datasets, reducing annotation costs and enabling scalable, fully automated CRLM segmentation. Additionally, MONAI offers an abundance of reusable components, from preprocessing tools to model training networks that were not evaluated within this project and could be further explored.

Better hardware capabilities could support more thorough experiments with larger datasets and improved quality. Upgraded GPUs with larger memory capabilities like 24–48 GB RTX or A100 GPUs could accelerate training and allow experimentation with increased batches and spatial sizes. Multi-GPU training or subscription programs from cloud-based platforms (Google Cloud, AWS, Azure) could provide infrastructure for distributed training on large datasets. These hardware upgrades could result in better and more stable training performance, experiments with deeper transformer-based architectures without memory fragmentation, and systematic hyperparameter tuning.

Overall, to create clinically reliable tools for CRLM segmentation, with potential application for medical facilities, advancements in computational power, augmentation strategies, algorithmic design, and dataset diversity are required.

Bibliography

[1] National Cancer Institute, "Seer cancer stat facts: Colorectal cancer." `https://seer.cancer.gov/statfacts/html/colorect.html`, 2025. Accessed: 2025-10-13.

[2] H. Tsutsui and S. Nishiguchi, "Importance of kupffer cells in the development of acute liver injuries in mice," *International Journal of Molecular Sciences*, vol. 15, no. 5, pp. 7711–7730, 2014.

[3] M. Pérez-Enciso and L. Zingaretti, "A guide on deep learning for complex trait genomic prediction," *Genes*, vol. 10, p. 553, July 2019.

[4] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, pp. 943–947, Nov. 2015.

[5] K. Y. Lim, J. E. Ko, Y. N. Hwang, S. G. Lee, and S. M. Kim, "Transraunet: A deep neural network with reverse attention module using hu windowing augmentation for robust liver vessel segmentation in full resolution of ct images," *Diagnostics*, vol. 15, no. 2, p. 118, 2025.

[6] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, pp. 234–241, 2015.

[7] A. Myronenko, "3d MRI brain tumor segmentation using autoencoder regularization," *arXiv preprint*, 2018. [Online]. Available: `https://arxiv.org/abs/1810.11654`.

[8] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. R. Roth, and D. Xu, "Unetr: Transformers for 3d medical image segmentation," *Proceedings - 2022 IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022*, pp. 1748–1758, Mar. 2021.

[9] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. Roth, and D. Xu, "Swin UNETR: Swin transformers for semantic segmentation of brain tumors in MRI images," *arXiv preprint*, 2022. [Online]. Available: `https://arxiv.org/abs/2201.01266`.

[10] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert, "Attention U-Net: Learning where to look for the pancreas," *arXiv preprint*, 2018. [Online]. Available: `https://arxiv.org/abs/1804.03999`.

[11] A. L. Simpson, J. Peoples, J. M. Creasy, G. Fichtinger, N. Gangai, K. N. Keshavamurthy, A. Lasso, J. Shia, M. I. D'Angelica, and R. K. Do, "Preoperative ct and survival data for patients undergoing resection of colorectal liver metastases," *Scientific Data 2024 11:1*, vol. 11, pp. 1–6, Feb. 2024.

[12] K. Simon, "Colorectal cancer development and advances in screening," *Clin. Interv. Aging*, vol. 11, pp. 967–976, July 2016.

[13] D. I. Tsilimigras, P. Brodt, P. A. Clavien, R. J. Muschel, M. I. D'Angelica, I. Endo, R. W. Parks, M. Doyle, E. de Santibañes, and T. M. Pawlik, "Liver metastases," *Nature Reviews Disease Primers*, vol. 7, pp. 1–23, Dec. 2021.

[14] E. Dekker, P. J. Tanis, J. L. Vleugels, P. M. Kasi, and M. B. Wallace, "Colorectal cancer," *The Lancet*, vol. 394, pp. 1467–1480, Oct. 2019.

[15] B. Levin, D. A. Lieberman, B. McFarland, K. S. Andrews, D. Brooks, J. Bond, C. Dash, F. M. Giardiello, S. Glick, D. Johnson, C. D. Johnson, T. R. Levin, P. J. Pickhardt, D. K. Rex, R. A. Smith, A. Thorson, and S. J. Winawer, "Screening and surveillance for the early detection of colorectal cancer and adenomatous polyps, 2008: A joint guideline from the american cancer society, the us multi-society task force on colorectal cancer, and the american college of radiology," *Gastroenterology*, vol. 134, pp. 1570–1595, 2008.

[16] R. E. Schoen, A. Razzak, K. J. Yu, S. I. Berndt, K. Firl, T. L. Riley, and P. F. Pinsky, "Incidence and mortality of colorectal cancer in individuals with a family history of colorectal cancer," *Gastroenterology*, vol. 149, pp. 1438–1445.e1, Nov. 2015.

[17] Colorectal Cancer Alliance, "Stages of colorectal cancer." [Online]. Accessed: May 18, 2025.

[18] T. André, R. Cohen, and M. E. Salem, "Immune checkpoint blockade therapy in patients with colorectal cancer harboring microsatellite instability/mismatch repair deficiency in 2022," *American Society of Clinical Oncology Educational Book*, pp. 233–241, July 2022.

[19] L. H. Biller and D. Schrag, "Diagnosis and treatment of metastatic colorectal cancer: A review," *JAMA - Journal of the American Medical Association*, vol. 325, pp. 669–685, Feb. 2021.

[20] A. E. Shin, F. G. Giancotti, and A. K. Rustgi, "Metastatic colorectal cancer: Mechanisms and emerging therapeutics," *Trends In Pharmacological Sciences*, vol. 44, p. 222, Apr. 2023.

[21] D. I. Tsilimigras, I. Ntanasis-Stathopoulos, and T. M. Pawlik, "Molecular mechanisms of colorectal liver metastases," *Cells*, vol. 12, June 2023.

[22] D. V. Tauriello and E. Batlle, "Targeting the microenvironment in advanced colorectal cancer," *Trends in Cancer*, vol. 2, pp. 495–504, Sept. 2016.

[23] K. Nagao, A. Koshino, A. Sugimura-Nagata, A. Nagano, M. Komura, A. Ueki, M. Ebi, N. Ogasawara, T. Tsuzuki, K. Kasai, S. Takahashi, K. Kasugai, and S. Inaguma, "The complete loss of p53 expression uniquely predicts worse prognosis in colorectal cancer," *International Journal of Molecular Sciences*, vol. 23, Mar. 2022.

[24] D. H. Ballard, K. R. Burton, N. Lakomkin, S. Kim, P. Rajiah, M. J. Patel, P. Mazaheri, and G. J. Whitman, "The role of imaging in health screening: overview, rationale of screening, and screening economics," *Academic Radiology*, vol. 28, p. 540, Apr. 2020.

[25] R. A. Smith, K. S. Andrews, D. Brooks, S. A. Fedewa, D. Manassaram-Baptiste, D. Saslow, and R. C. Wender, "Cancer screening in the united states, 2019: A review of current american cancer society guidelines and current issues in cancer screening," *CA: A Cancer Journal for Clinicians*, vol. 69, pp. 184–210, May 2019.

[26] M. F. Kaminski, J. Regula, E. Kraszewska, M. Polkowski, U. Wojciechowska, J. Didkowska, M. Zwierko, M. Rupinski, M. P. Nowacki, and E. Butruk, "Quality indicators for colonoscopy and the risk of interval cancer," *New England Journal of Medicine*, vol. 362, pp. 1795–1803, May 2010.

[27] B. Lauby-Secretan, N. Vilahur, F. Bianchini, N. Guha, and K. Straif, "The iarc perspective on colorectal cancer screening," *New England Journal of Medicine*, vol. 378, pp. 1734–1740, May 2018.

[28] A. Mitsala, C. Tsalikidis, M. Pitiakoudis, C. Simopoulos, and A. K. Tsaroucha, "Artificial intelligence in colorectal cancer screening, diagnosis and treatment. a new era," *Current Oncology*, vol. 28, p. 1581, 2021.

[29] L. Spelt, B. Andersson, J. Nilsson, and R. Andersson, "Prognostic models for outcome following liver resection for colorectal cancer metastases: A systematic review," *European Journal of Surgical Oncology*, vol. 38, pp. 16–24, Jan. 2012.

[30] B. Hazhirkarzar, P. Khoshpouri, M. Shaghaghi, M. A. Ghasabeh, T. M. Pawlik, and I. R. Kamel, "Current state of the art imaging approaches for colorectal liver metastasis," *Hepatobiliary Surgery and Nutrition*, vol. 9, pp. 348–358, Feb. 2020.

[31] D. Maclean, M. Tsakok, F. Gleeson, D. J. Breen, R. Goldin, J. Primrose, A. Harris, and J. Franklin, "Comprehensive imaging characterization of colorectal liver metastases," *Frontiers in Oncology*, vol. 11, p. 730854, Dec. 2021.

[32] A. Kumar, V. Gautam, A. Sandhu, K. Rawat, A. Sharma, and L. Saha, "Current and emerging therapeutic approaches for colorectal cancer: A comprehensive review," *World Journal of Gastrointestinal Surgery*, vol. 15, pp. 495–519, Apr. 2023.

[33] C. A. Thiels and M. I. D'Angelica, "Hepatic artery infusion pumps," *Journal of Surgical Oncology*, vol. 122, p. 70, July 2020.

[34] E. Al-Sharif, E. Simoneau, and M. Hassanain, "Portal vein embolization effect on colorectal cancer liver metastasis progression: Lessons learned," *World Journal of Clinical Oncology*, vol. 6, p. 142, Oct. 2015.

[35] A. R. Townsend, L. C. Chong, C. Karapetis, and T. J. Price, "Selective internal radiation therapy for liver metastases from colorectal cancer," *Cancer Treatment Reviews*, vol. 50, pp. 148–154, Nov. 2016.

[36] I. Mohamad, A. Barry, L. Dawson, and A. Hosni, "Stereotactic body radiation therapy for colorectal liver metastases," *International Journal of Hyperthermia*, vol. 39, pp. 611–619, Dec. 2022.

[37] A. Mimmo, F. Pegoraro, R. Rhaiem, R. Montalti, A. Donadieu, A. Tashkandi, A. R. Al-Sadairi, R. Kianmanesh, and T. Piardi, "Microwave ablation for colorectal liver metastases: A systematic review and pooled oncological analyses," *Cancers*, vol. 14, p. 1305, Mar. 2022.

[38] A. Stoltz, J. Gagnière, A. Dupré, and M. Rivoire, "Radiofrequency ablation for colorectal liver metastases," *Journal of Visceral Surgery*, vol. 151, pp. S33–S44, Apr. 2014.

[39] A. Subasi, *Machine learning techniques*, pp. 91–202. Academic Press, Jan. 2020.

[40] S. Sarkar, P. Teo, and M. Abazeed, "Deep learning for automated, motion-resolved tumor segmentation in radiotherapy," *npj Precision Oncology*, vol. 9, p. 173, 2025. Published 30 June 2025; Received 14 January 2025; Accepted 27 May 2025.

[41] R. Alalwani, A. Lucas, M. Alzubaidi, H. A. Shah, T. Alam, Z. Shah, and M. Househ, "Deep learning in colorectal cancer classification: A scoping review," *Studies in health technology and informatics*, vol. 305, pp. 616–619, June 2023.

[42] T. Dhar, N. Dey, S. Borra, and R. Sherratt, "Challenges of deep learning in medical image analysis: Improving explainability and trust," *IEEE Transactions on Technology and Society*, vol. PP, pp. 1–1, mar 2023.

[43] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Adaptive Computation and Machine Learning series, Cambridge, MA, USA: The MIT Press, 2012.

[44] J. G. Lee, S. Jun, Y. W. Cho, H. Lee, G. B. Kim, J. B. Seo, and N. Kim, "Deep learning in medical imaging: General overview," *Korean Journal of Radiology*, vol. 18, p. 570, 2017.

[45] A. Anaya-Isaza, L. Mera-Jiménez, and M. Zequera-Diaz, "An overview of deep learning in medical imaging," *Informatics in Medicine Unlocked*, vol. 26, p. 100723, Jan. 2021.

[46] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[47] A. Aljuaid and M. Anwar, "Survey of supervised learning for medical image processing," *SN Computer Science*, vol. 3, pp. 1–22, July 2022.

[48] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, pp. 373–440, Feb. 2020.

[49] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[50] J. Ramírez-Sanz, J.-A. Maestro-Prieto, Á. Arnaiz-González, and A. Bustillo, "Semi-supervised learning for industrial fault detection and diagnosis: A systemic review," *ISA Transactions*, vol. 143, pp. —, Sept. 2023.

[51] A. Arora, R. Jordar, J. J. Jena, S. Singh, S. S. Patra, and M. K. Gourisaria, "Harnessing deep learning and transfer learning models for segmentation of liver tumors and veins," *Procedia Computer Science*, vol. 259, pp. 1874–1882, Jan. 2025.

[52] J. Wacker, M. Ladeira, and J. E. V. Nascimento, "Transfer learning for brain tumor segmentation," *arXiv preprint*, 2020. [Online]. Available: `https://arxiv.org/abs/1912.12452`.

[53] M. Islam, G. Chen, and S. Jin, "An overview of neural network," *American Journal of Neural Networks and Applications*, vol. 5, no. 1, pp. 7–11, 2019.

[54] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sept. 2021.

[55] M. Pandey, M. Fernandez, F. Gentile, O. Isayev, A. Tropsha, A. C. Stern, and A. Cherkasov, "The transformational role of GPU computing and deep learning in drug discovery," *Nature Machine Intelligence*, vol. 4, pp. 211–221, Mar 2022.

[56] S.-H. Noh, "Analysis of gradient vanishing of rnns and performance comparison," *Information*, vol. 12, no. 11, p. 442, 2021.

[57] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *arXiv preprint*, 2015. [Online]. Available: `https://arxiv.org/abs/1411.4038`.

[58] M. H. Hesamian, W. Jia, X. He, and P. Kennedy, "Deep learning techniques for medical image segmentation: Achievements and challenges," *Journal of Digital Imaging*, vol. 32, pp. 582–596, Aug. 2019.

[59] C. Chen, N. A. M. Isa, and X. Liu, "A review of convolutional neural network based methods for medical image classification," *Computers in Biology and Medicine*, vol. 185, p. 109507, Feb. 2025.

[60] R. Wang, T. Lei, R. Cui, B. Zhang, H. Meng, and A. K. Nandi, "Medical image segmentation using deep learning: A survey," *IET Image Processing*, vol. 16, pp. 1243–1267, Apr. 2022.

[61] P. Bilic, P. Christ, H. B. Li, E. Vorontsov, A. Ben-Cohen, G. Kaissis, A. Szeskin, C. Jacobs, G. E. H. Mamani, G. Chartrand, F. Lohöfer, J. W. Holch, W. Sommer, F. Hofmann, A. Hostettler, N. Lev-Cohain, M. Drozdzal, M. M. Amitai, R. Vivanti, J. Sosna, I. Ezhov, A. Sekuboyina, F. Navarro, F. Kofler, J. C. Paetzold, S. Shit, X. Hu, J. Lipková, M. Rempfler, M. Piraud, J. Kirschke, B. Wiestler, Z. Zhang, C. Hülsemeyer, M. Beetz, F. Ettlinger, M. Antonelli, W. Bae, M. Bellver, L. Bi, H. Chen, G. Chlebus, E. B. Dam, Q. Dou, C. W. Fu, B. Georgescu,

X. G. i Nieto, F. Gruen, X. Han, P. A. Heng, J. Hesser, J. H. Moltz, C. Igel, F. Isensee, P. Jäger, F. Jia, K. C. Kaluva, M. Khened, I. Kim, J. H. Kim, S. Kim, S. Kohl, T. Konopczynski, A. Kori, G. Krishnamurthi, F. Li, H. Li, J. Li, X. Li, J. Lowengrub, J. Ma, K. Maier-Hein, K. K. Maninis, H. Meine, D. Merhof, A. Pai, M. Perslev, J. Petersen, J. Pont-Tuset, J. Qi, X. Qi, O. Rippel, K. Roth, I. Sarasua, A. Schenk, Z. Shen, J. Torres, C. Wachinger, C. Wang, L. Weninger, J. Wu, D. Xu, X. Yang, S. C. H. Yu, Y. Yuan, M. Yue, L. Zhang, J. Cardoso, S. Bakas, R. Braren, V. Heinemann, C. Pal, A. Tang, S. Kadoury, L. Soler, B. van Ginneken, H. Greenspan, L. Joskowicz, and B. Menze, "The liver tumor segmentation benchmark (lits)," *Medical Image Analysis*, vol. 84, p. 102680, Feb. 2023.

[62] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[63] C. Wu, Q. Chen, H. Wang, Y. Guan, Z. Mian, C. Huang, C. Ruan, Q. Song, H. Jiang, J. Pan, and X. Li, "A review of deep learning approaches for multimodal image segmentation of liver cancer," *Journal of Applied Clinical Medical Physics*, 2024.

[64] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[65] B. Hussein and S. Shareef, "An empirical study on the correlation between early stopping patience and epochs in deep learning," *ITM Web of Conferences*, vol. 64, p. 01003, July 2024.

[66] O. Elharrouss, Y. Mahmood, Y. Bechqito, M. A. Serhani, E. Badidi, J. Riffi, and H. Tairi, "Loss functions in deep learning: A comprehensive review," *arXiv preprint arXiv:2504.04242*, 2025.

[67] M. Cabezas and Y. Diez, "An analysis of loss functions for heavily imbalanced lesion segmentation," *Sensors (Basel, Switzerland)*, vol. 24, p. 1981, Mar. 2024.

[68] K. Institute, "Anything but SGD: Evaluating optimizers for LLM training." `https://kempnerinstitute.harvard.edu/research/deeper-learning/anything-but-sgd-evaluating-optimizers-for-llm-training/`, 2024. Accessed: 2025-08-19.

[69] A. Gupta, R. Ramanath, J. Shi, and S. S. Keerthi, "Adam vs. SGD: Closing the generalization gap on image classification," in *Proceedings of the 13th Annual Workshop on Optimization for Machine Learning (OPT 2021)*, (Virtual Conference), LinkedIn, Sunnyvale, CA, 2021.

[70] R. Meshaka and O. J. Arthurs, "Are we too reliant on medical imaging?," *https://doi.org/10.12968/hmed.2022.0460*, vol. 83, Dec. 2022.

[71] Y. Xu, R. Quan, W. Xu, Y. Huang, X. Chen, and F. Liu, "Advances in medical image segmentation: A comprehensive review of traditional, deep learning and hybrid approaches," *Bioengineering*, vol. 11, no. 10, p. 1034, 2024.

[72] E. Abbaspour, S. Karimzadhagh, A. Monsef, F. Joukar, F. Mansour-Ghanaei, and S. Hassanipour, "Application of radiomics for preoperative prediction of lymph node metastasis in colorectal cancer: a systematic review and meta-analysis," *International journal of surgery (London, England)*, vol. 110, pp. 3795–3813, June 2024.

[73] P. K. Mall, P. K. Singh, S. Srivastav, V. Narayan, M. Paprzycki, T. Jaworska, and M. Ganzha, "A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities," *Healthcare Analytics*, vol. 4, p. 100216, Dec. 2023.

[74] I. Pacal, D. Karaboga, A. Basturk, B. Akay, and U. Nalbantoglu, "A comprehensive review of deep learning in colon cancer," *Computers in Biology and Medicine*, vol. 126, p. 104003, 2020.

[75] W. Bi, A. Hosny, M. Schabath, M. Giger, N. Birkbak, A. Mehrtash, T. Allison, O. Arnaout, C. Abbosh, I. Dunn, R. Mak, R. Tamimi, C. Tempany, C. Swanton, U. Hoffmann, L. Schwartz, R. Gillies, R. Huang, and H. Aerts, "Artificial intelligence in cancer imaging: Clinical challenges and applications," *CA: A Cancer Journal for Clinicians*, vol. 69, pp. 127–157, mar 2019. Epub 2019 Feb 5.

[76] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, "A review of semantic segmentation using deep neural networks," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 87–93, 2018.

[77] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, HI, USA), pp. 876–885, 2017.

[78] S. K. M. S. Islam, M. D. A. A. Nasim, I. Hossain, M. A. Ullah, K. D. Gupta, and M. M. H. Bhuiyan, "Introduction of medical imaging modalities," *arXiv preprint*, Aug. 2024. [Online]. Available: `https://arxiv.org/abs/2306.01022`.

[79] F. Galbusera and A. Cina, "Image annotation and curation in radiology: an overview for machine learning practitioners," *European Radiology Experimental*, vol. 8, p. 11, Dec. 2024.

[80] A. Avesta, S. Hossain, M. Lin, M. Aboian, H. M. Krumholz, and S. Aneja, "Comparing 3d, 2.5d, and 2d approaches to brain image auto-segmentation," *Bioengineering (Basel, Switzerland)*, vol. 10, no. 2, p. 181, 2023.

[81] N. Zettler and A. Mastmeyer, "Comparison of 2d vs. 3d u-net organ segmentation in abdominal 3d ct images," *Computer Science Research Notes*, vol. 3101, pp. 41–50, July 2021.

[82] B. M. Tummala and S. S. Barpanda, "Curriculum learning based overcomplete u-net for liver tumor segmentation from computed tomography images," *Bulletin of Electrical Engineering and Informatics*, vol. 12, pp. 1620–1629, June 2023.

[83] M. E. A. Elforaici, F. Azzi, D. Trudel, B. Nguyen, E. Montagnon, A. Tang, S. Turcotte, and S. Kadoury, "Cell-level gnn-based prediction of tumor regression grade in colorectal liver metastases from histopathology images," *Proceedings - International Symposium on Biomedical Imaging*, 2024.

[84] V. T. T. Vo, H. J. Yang, G. S. Lee, S. R. Kang, and S. H. Kim, "Effects of multiple filters on liver tumor segmentation from ct images," *Frontiers in Oncology*, vol. 11, Oct. 2021.

[85] S. Muhammad and J. Zhang, "Segmentation of liver tumors by monai and pytorch in ct images with deep learning techniques," *Applied Sciences*, vol. 14, no. 12, p. 5144, 2024.

[86] K. Kim and J. Chun, "A new hyper parameter of hounsfield unit range in liver segmentation," *Journal of Internet Computing and Services*, vol. 21, pp. 103–111, 2020.

[87] M. S. A. Magboo and A. D. Coronel, "Effects of cropping vs resizing on the performance of brain tumor segmentation models," *Proceedings of the 2024 IEEE International Conference on Computer, Information, and Telecommunication Systems, CITS 2024*, 2024.

[88] R. R. Outeiral, P. Bos, H. J. van der Hulst, A. Al-Mamgani, B. Jasperse, R. Simões, and U. A. van der Heide, "Strategies for tackling the class imbalance problem of oropharyngeal primary tumor segmentation on magnetic resonance imaging," *Physics and Imaging in Radiation Oncology*, vol. 23, p. 144, July 2022.

[89] Z. Li, K. Kamnitsas, and B. Glocker, "Analyzing overfitting under class imbalance in neural networks for image segmentation," *IEEE Transactions on Medical Imaging*, vol. 40, pp. 1065–1077, Mar. 2021.

[90] Özgün Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: Learning dense volumetric segmentation from sparse annotation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9901 LNCS, pp. 424–432, 2016.

[91] H. Rahman, T. F. N. Bukht, A. Imran, J. Tariq, S. Tu, and A. Alzahrani, "A deep learning approach for liver and tumor segmentation in ct images using resunet," *Bioengineering*, vol. 9, Aug. 2022.

[92] K. S. Sheela, V. Justus, R. R. Asaad, and R. L. Kumar, "Enhancing liver tumor segmentation with unet-resnet: Leveraging resnet's power," *Technology and health care : official journal of the European Society for Engineering and Medicine*, pp. 1–15, Sept. 2024.

[93] G. Hille, S. Agrawal, P. Tummala, C. Wybranski, M. Pech, A. Surov, and S. Saalfeld, "Joint liver and hepatic lesion segmentation in mri using a hybrid cnn with transformer layers," *Computer Methods and Programs in Biomedicine*, vol. 240, Oct. 2023.

[94] P. F. Christ, F. Ettlinger, F. Grün, M. Ezzeldin, A. Elshaer, J. Lipková, S. Schlecht, F. Ahmaddy, S. Tatavarty, M. Bickel, P. Bilic, M. Rempfler, F. Hofmann, M. D'anastasi, S.-A. Ahmadi, G. Kaissis, J. Holch, W. Sommer, R. Braren, V. Heinemann, and B. Menze, "Automatic liver and tumor segmentation of ct and mri volumes using cascaded fully convolutional neural networks," *arXiv preprint*, Feb. 2017. [Online]. Available: `https://arxiv.org/pdf/1702.05970`.

[95] F. Milletari, N. Navab, and S. A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016*, pp. 565–571, Dec. 2016.

[96] harrigr, "Segmentation outside the cranial vault challenge," 2015. Accessed: 2025-08-26.

[97] M. Antonelli, A. Reinke, S. Bakas, K. Farahani, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze, O. Ronneberger, R. M. Summers, B. van Ginneken, M. Bilello, P. Bilic, P. F. Christ, R. K. G. Do, M. J. Gollub, S. H. Heckers, H. Huisman, W. R. Jarnagin, M. K. McHugo, S. Napel, J. S. G. Pernicka, K. Rhode, C. Tobon-Gomez, E. Vorontsov, J. A. Meakin, S. Ourselin, M. Wiesenfarth, P. Arbeláez, B. Bae, S. Chen, L. Daza, J. Feng, B. He, F. Isensee, Y. Ji, F. Jia, I. Kim, K. Maier-Hein, D. Merhof, A. Pai, B. Park, M. Perslev, R. Rezaiifar, O. Rippel, I. Sarasua, W. Shen, J. Son, C. Wachinger, L. Wang, Y. Wang, Y. Xia, D. Xu, Z. Xu, Y. Zheng, A. L. Simpson, L. Maier-Hein, and M. J. Cardoso, "The medical segmentation decathlon," *Nature Communications*, vol. 13, no. 1, p. 4128, 2022.

[98] A. E. Kavur, N. S. Gezer, M. Barış, S. Aslan, P.-H. Conze, V. Groza, D. D. Pham, S. Chatterjee, P. Ernst, S. Özkan, B. Baydar, D. Lachinov, S. Han, J. Pauli, F. Isensee, M. Perkonigg, R. Sathish, R. Rajan, D. Sheet, G. Dovletov, O. Speck, A. Nürnberger, K. H. Maier-Hein, G. B. Akar, G. Ünal, O. Dicle, and M. A. Selver, "Chaos challenge – combined (ct-mr) healthy abdominal organ segmentation," *Medical Image Analysis*, vol. 69, p. 101950, 2021.

[99] J. Huiskens, T. M. van Gulik, K. P. van Lienden, M. R. W. Engelbrecht, G. A. Meijer, N. C. T. van Grieken, J. Schriek, A. Keijser, L. Mol, I. Q. Molenaar, C. Verhoef, K. P. de Jong, K. H. C. Dejong, G. Kazemier, T. M. Ruers, J. H. W. de Wilt, H. van Tinteren, and C. J. A. Punt, "Treatment strategies in colorectal cancer patients with initially unresectable liver-only metastases: A study protocol of the randomised phase 3 cairo5 study of the dutch colorectal cancer group (dccg)," *BMC Cancer*, vol. 15, no. 1, p. 365, 2015.

[100] B. M. Anderson, B. Rigaud, Y. M. Lin, A. K. Jones, H. S. C. Kang, B. C. Odisio, and K. K. Brock, "Automated segmentation of colorectal liver metastasis and liver ablation on contrast-enhanced ct images," *Frontiers in Oncology*, vol. 12, p. 886517, Aug. 2022.

[101] F. Özcan, O. N. Uçan, S. Karaçam, and D. Tunçman, "Fully automatic liver and tumor segmentation from ct image using an aim-unet.," *Bioengineering (Basel, Switzerland)*, vol. 10, Feb. 2023.

[102] G. N. Yashaswini, R. V. Manjunath, B. Shubha, P. Prabha, N. Aishwarya, and H. M. Manu, "Deep learning technique for automatic liver and liver tumor segmentation in ct images," *Journal of Liver Transplantation*, vol. 17, p. 100251, Feb. 2025.

[103] J. Wang, Y. Peng, S. Jing, L. Han, T. Li, and J. Luo, "A deep-learning approach for segmentation of liver tumors in magnetic resonance imaging using unet++," *BMC Cancer*, vol. 23, 2023.

[104] N. J. Wesdorp, J. M. Zeeuw, S. C. Postma, J. Roor, J. H. T. van Waesberghe, J. E. van den Bergh, I. M. Nota, S. Moos, R. Kemna, F. Vadakkumpadan, C. Ambrozic, S. van Dieren, M. J. van Amerongen, T. Chapelle, M. R. Engelbrecht, M. F. Gerhards, D. Grunhagen, T. M. van Gulik, J. J. Hermans, K. P. de Jong, J. M. Klaase, M. S. Liem, K. P. van Lienden, I. Q. Molenaar, G. A. Patijn, A. M. Rijken, T. M. Ruers, C. Verhoef, J. H. de Wilt, H. A. Marquering, J. Stoker, R. J. Swijnenburg, C. J. Punt, J. Huiskens, and G. Kazemier, "Deep learning models for automatic tumor segmentation and total tumor volume assessment in patients with colorectal liver metastases," *European radiology experimental*, vol. 7, Dec. 2023.

[105] K. He, X. Liu, R. Shahzad, R. Reimer, F. Thiele, J. Niehoff, C. Wybranski, A. C. Bunck, H. Zhang, and M. Perkuhn, "Advanced deep learning approach to automatically segment malignant tumors and ablation zone in the liver with contrast-enhanced ct," *Frontiers in Oncology*, vol. 11, p. 669437, July 2021.

[106] J. I. Bereska, M. Zeeuw, L. Wagenaar, H. B. Jenssen, N. J. Wesdorp, D. van der Meulen, L. F. Bereska, E. Gavves, B. V. Janssen, M. G. Besselink, H. A. Marquering, J. H. T. van Waesberghe, D. L. Aghayan, E. Pelanis, J. van den Bergh, I. I. Nota, S. Moos, G. Kemmerich, T. Syversveen, F. K. Kolrud, J. Huiskens, R. J. Swijnenburg, C. J. Punt, J. Stoker, B. Edwin, Åsmund A. Fretland, G. Kazemier, I. M. Verpalen, C. Michalski, M. Loos, B. Kinny-Köster, P. Mayer, M. D. Pomohaci, C. Anghel, C. M. Grasu, I. Lupescu, T. Stoop, T. Clark, J. Kaplan, M. D. Chiaro, K. Colborn, A. Javed, C. Wolfgang, R. Salvia, G. Malleo, A. Balduzzi, C. Luchini, R. di Robertis, G. Zamboni, M. D'Onofrio, Åsmund Fretland, K. J. Labori, C. Verbeke, A. Farina, M. Fassan, F. Crimi, R. Carandina, M. Ballo, R. Boetto, D. Bassi, G. Marchegiani, J. H. de Wilt, C. Verhoef, T. M. Ruers, A. M. Rijken, G. A. Patijn, I. Q. Molenaar, K. P. van Lienden, M. S. Liem, W. K. Leclercq, N. F. Kok, J. M. Klaase, K. P. de Jong, J. J. Hermans, T. M. van Gulik, D. J. Grunhagen, M. F. Gerhards, M. R. Engelbrecht, R. M. van Dam, T. Chapelle, M. J. Bond, and M. J. van Amerongen, "Development and external evaluation of a self-learning auto-segmentation model for colorectal cancer liver metastases assessment (coala)," *Insights into Imaging*, vol. 15, p. 279, Dec. 2024.

[107] R. Manjunath and K. Kwadiki, "Modified u-net on ct images for automatic segmentation of liver and its tumor," *Biomedical Engineering Advances*, vol. 4, p. 100043, Dec. 2022.

[108] M. E. A. Mokhtari, "Liver segmentation using monai and pytorch," 2024.

[109] J. Li, K. Liu, Y. Hu, H. Zhang, A. A. Heidari, H. Chen, W. Zhang, A. D. Algarni, and H. Elmannai, "Eres-unet++: Liver ct image segmentation based on high-

efficiency channel attention and res-unet++," *Computers in Biology and Medicine*, vol. 158, May 2023.

[110] V. Boussot and J.-L. Dillenseger, "Konfai: A modular and fully configurable framework for deep learning in medical imaging," *arXiv preprint*, 2025. [Online]. Available: `https://arxiv.org/abs/2508.09823`.

[111] J. Becktepe, L. Hennig, S. Oeltze-Jafra, and M. Lindauer, "Auto-nnu-net: Towards automated medical image segmentation," *arXiv preprint*, 2025. [Online]. Available: `https://arxiv.org/abs/2505.16561v1`.

Appendix A

Model implementation listings

Listing 1: UNETR for liver/tumor segmentation

```
from monai.networks.nets import UNETR

model = UNETR(
    in_channels=1,
    out_channels=2,
    img_size=(128, 128, 64),
    feature_size=32,
    hidden_size=768,
    mlp_dim=3072,
    num_heads=12, # multi-head self-attention
    pos_embed="perceptron",
    norm_name="instance",
    res_block=True,
    dropout_rate=0.1,
)
```

Listing 2: Swin UNETR (shifted-window ViT backbone)

```
from monai.networks.nets import SwinUNETR

model = SwinUNETR(
    img_size=(96, 96, 32), # NOTE: each dim should be divisible by
       32
    in_channels=1,
    out_channels=2,
    feature_size=48,
    depths=(2, 2, 2, 2),
    num_heads=(3, 6, 12, 24),
    norm_name="instance",
    drop_rate=0.1,
    attn_drop_rate=0.0,
    dropout_path_rate=0.0,
    normalize=True,
    use_checkpoint=False,
    spatial_dims=3,
    downsample="merging",
    use_v2=False
)
```

Listing 3: Attention U-Net (3D)

```
from monai.networks.nets import AttentionUnet

model = AttentionUnet(
    spatial_dims=3,
    in_channels=1,
```

```
    out_channels=2,
    channels=(16, 32, 64, 128, 256),
    strides=((2,2,1),(2,2,1),(2,2,2),(2,2,2)), # 4 downsampling
       levels
    kernel_size=3,
    up_kernel_size=3,
    dropout=0.1
)
```

Listing 4: U-Net (ResUNet-like, residual units + BN)

```
from monai.networks.nets import UNet

model = UNet(
    spatial_dims=3,
    in_channels=1,
    out_channels=2,
    channels=(16, 32, 64, 128, 256),
    strides=((2,2,1),(2,2,1),(2,2,2),(2,2,2)),
    num_res_units=2,
    act='PRELU',
    norm='INSTANCE',
    dropout=0.1
)
```

Listing 5: SegResNet (3D) for robust CT segmentation

```
from monai.networks.nets import SegResNet

model = SegResNet(
    spatial_dims=3,
    init_filters=16,
    in_channels=1,
    out_channels=2,
    dropout_prob=0.1,
    act=('RELU', {'inplace': True})
    norm=('GROUP', {'num_groups': 8}),
    num_groups=8,
    use_conv_final=True,
    blocks_down=(1, 2, 2, 4),
    blocks_up=(1, 1, 1),
)
```