UFR DE MÉDECINE
ET DES PROFESSIONS PARAMÉDICALES
Université Clermont Auvergne

DOCTORAL SCHOOL OF ENGINEERING SCIENCES

THESIS

to obtain the degree of Doctor awarded by

Clermont Auvergne University

(Decree of July 5, 1984)

Speciality
Computer Vision

presented and publicly defended by

Ivan MIKHAILOV

on 28 March 2025

Interactive Segmentation and its Use at Scale through Concurrent Annotation and Training

Supervisor: **Adrien BARTOLI**
Co-supervisor: **Nicolas BOURDEL**

Adrian BASARAB,	Professor, University of Lyon, President of the Jury, Examiner
Nicolas THOME,	Professor, Sorbonne University, Reviewer
Elsa ANGELINI,	Professor, Télécom Paris, Reviewer
Benoit CHAUVEAU,	Radiologist, Clermont-Ferrand University Hospital, Invited Member

UFR DE MÉDECINE
ET DES PROFESSIONS PARAMÉDICALES
Université Clermont Auvergne

ÉCOLE DOCTORALE DES SCIENCES POUR L'INGÉNIEUR

THÈSE

pour obtenir le grade de Docteur délivré par

l'Université Clermont Auvergne

(Décret du 5 Juillet 1984)

Spécialité
Vision par Ordinateur

présentée et soutenue publiquement par

Ivan MIKHAILOV

le 28 Mars, 2025

Segmentation interactive et son utilisation à grande échelle par annotation et entraînement simultanés

Directeur de thèse : **Adrien BARTOLI**
Co-encadrant de thèse : **Nicolas BOURDEL**

Adrian BASARAB,	Professeur, Université de Lyon, Président du Jury, Examinateur
Nicolas THOME,	Professeur, Sorbonne Université, Rapporteur
Elsa ANGELINI,	Professeur, Télécom Paris, Rapporteur
Benoit CHAUVEAU,	Radiologue, CHU Clermont-Ferrand, Invité

EnCoV Institut Pascal
SURGAR Surgical Augmented Reality

Abstract

Machine Learning (ML) is a field of study focused on developing statistical algorithms that enable systems to learn from data and make predictions or decisions without explicitly programmed instructions. These algorithms are embodied in models, which represent the learned patterns and relationships within the data. The learning process, typically referred to as training, involves adjusting the model's parameters based on the input data. Once trained, the model can be utilized to make predictions or decisions on new, unseen data. Data annotation is a cornerstone of ML, as it involves creating ground truth labels, descriptive categories or markers assigned to each data point, that guide the training process. These annotations help ensure that the model learns meaningful information during training, leading to better predictions. An ML model is often an Artificial Neural Network (ANN), which consists of layers of connected artificial neurons. Deep Learning (DL) is a subset of ML that utilizes deep networks, characterized by an increased number of layers, which often results in superior performance in exchange for a larger training dataset required.

The demand for large, annotated datasets is particularly pronounced in medical imaging, where accurate and efficient annotation remains a significant challenge due to the complexity of medical data, the need for domain expertise, and the critical implications of model predictions. Segmentation, a key task in medical image interpretation, exemplifies these challenges. Specifically, it requires precise per-pixel delineation of anatomical structures and pathologies normally done by a medical expert, which is time-consuming and scales poorly with the growth of the dataset. Medical image annotation thus poses two primary challenges: (1) the need for clinically-adapted annotation solutions, which speed up annotation and improve performance, and (2) the need for dedicated systems designed for annotation at scale to produce the training data for the latter. We outline the basis for each challenge in turn.

Medical image segmentation can be done manually, automatically, or using semi-automatic (interactive) segmentation algorithms. The resulting segmentations are typically used for diagnostics or a number of downstream tasks, such as to generate 3D models for surgical Augmented Reality (AR) as in this work. Given the high stakes of medical applications, interactive segmentation algorithms are more adapted to the clinical environment. This is due to the medical expert's involvement, who has the option to correct the segmentation proposed by the model, if necessary. Therefore, for efficient correction there is a need for the model to be able to interpret user corrections and infer their intention, which is not fully considered by existing approaches.

Efficient annotation systems typically utilize a neural annotation predictor that is initially trained on data produced by classical tools. This predictor then remains fixed throughout the annotation process. This is suboptimal for two key reasons: (1) the trained predictor should be already available, and (2) the predictor does not benefit from new annotations as the annotation process progresses. This means that extensive annotated data is required upfront to obtain an efficient annotation tool, which is especially challenging for tasks where the available annotated data is limited. An example is Female Pelvis MRI (FPMRI) segmentation, which is one of the key applications of this thesis. Simply, this creates a circular dependency where annotated data is required to produce more annotated data - an issue, which is not directly addressed in the literature.

In this thesis, we address the two primary challenges in medical image annotation by intro-

ducing four distinct contributions applied to segmentation specifically and annotation in general. These contributions are categorized into two data-centric and two application-centric approaches, with the data-centric contributions laying the foundation for the application-centric ones. On the data side, we introduced a new FPMRI segmentation dataset, Female Pelvis MRI dataset (FPMRId), and investigated its inter-expert variability. On the application side, we proposed two key solutions tailored for clinical and industrial use. The first addresses the need for clinically-adapted annotation tools, and the second provides a dedicated system for large-scale annotation. First, we propose a general framework for interactive segmentation not limited to a specific domain, which improves performance by incorporating the way the user typically approaches segmentation. Specifically, existing interactive segmentation methods do not utilise the sequential order of user interactions, as they typically disregard the sequence in which corrections are made. The proposed solution addresses this limitation by introducing an interaction memory, which preserves the order of user inputs, and incorporates it into the training. In this way, the model treats each correction based on previous inputs, resulting in higher accuracy with fewer interaction steps. Second, we proposed Single Active Interactive Model (SAIM), a framework that integrates data selection, annotation, and training into a unified architecture via model-sharing to enable efficient annotation at scale. SAIM is a 'two-in-one' solution, where the shared model is both an annotation tool and a predictor, ready to be deployed. SAIM operates on a loop. It uses the shared model pre-trained on a few hundreds of images to select the most informative data for annotation, suggest the annotations and interactively correct them, if necessary. The model is then updated with the produced annotations. This process is repeated until stopped or all available data is annotated. This approach mitigates the need for upfront availability of large annotated datasets. Specifically, it allows to start with limited annotated data and gradually improve the model, while reducing the annotation workload through data selection. At the same time, the model remains deployable for the task at any stage. Although SAIM is applied to medical imaging segmentation, it is not restricted to any specific task or domain.

Résumé

L'apprentissage automatique (Machine Learning, ML) est un domaine d'étude qui se concentre sur le développement d'algorithmes statistiques permettant aux systèmes d'apprendre à partir de données et de faire des prédictions ou de prendre des décisions sans recourir à des instructions explicitement programmées. Ces algorithmes se concrétisent sous forme de modèles, lesquels représentent les motifs et les relations appris au sein des données. Le processus d'apprentissage, appelé « entraînement », consiste à ajuster les paramètres du modèle en fonction des données d'entrée. Une fois entraîné, le modèle peut être employé pour effectuer des prédictions ou prendre des décisions sur de nouvelles données, encore jamais observées. L'annotation des données constitue une pierre angulaire de l'apprentissage automatique : elle consiste à associer, à chaque point de données, des étiquettes vérités terrain, des catégories descriptives ou des marqueurs. Ces annotations orientent le processus d'entraînement et aident à garantir que le modèle assimile des informations pertinentes, aboutissant à de meilleures prédictions. Un modèle d'apprentissage automatique est fréquemment un réseau de neurones artificiels, formé de couches de neurones artificiels interconnectés. L'apprentissage profond (Deep Learning, DL) est une sous-catégorie de l'apprentissage automatique qui exploite des réseaux dits « profonds », caractérisés par un nombre accru de couches, offrant souvent de meilleures performances, mais nécessitant en contrepartie un volume de données d'entraînement plus important.

La demande de grands ensembles de données annotées est particulièrement marquée en imagerie médicale, où l'annotation précise et efficace demeure un défi majeur en raison de la complexité des données, de la nécessité d'une expertise spécialisée et des implications critiques des prédictions du modèle. La segmentation, tâche clé dans l'interprétation des images médicales, illustre bien ces difficultés. En effet, elle exige une délinéation précise — pixel par pixel — des structures anatomiques et des pathologies, opération généralement réalisée par un expert médical, chronophage et mal adaptée à l'accroissement continu des ensembles de données. Ainsi, l'annotation en imagerie médicale soulève deux défis principaux: (1) la nécessité de solutions d'annotation cliniquement adaptées, capables d'accélérer l'annotation et d'en améliorer les performances, et (2) la nécessité de systèmes spécifiquement conçus pour l'annotation à grande échelle, afin de produire les données d'entraînement requises par ces solutions. Nous détaillons successivement les bases de chacun de ces défis.

La segmentation d'images médicales peut s'effectuer manuellement, automatiquement ou à l'aide d'algorithmes de segmentation semi-automatique (interactifs). Les segmentations obtenues sont généralement employées à des fins diagnostiques ou pour diverses tâches ultérieures, comme la génération de modèles 3D en réalité augmentée (RA) chirurgicale, à l'instar du travail présenté ici. Étant donné l'importance critique des applications médicales, les algorithmes de segmentation interactive sont mieux adaptés au contexte clinique. Cela s'explique par l'intervention de l'expert médical, qui peut, le cas échéant, corriger la segmentation proposée par le modèle. Par conséquent, pour optimiser ces corrections, il importe que le modèle sache interpréter les interventions de l'utilisateur et en déduire l'intention sous-jacente, aspect qui n'est pas pleinement pris en compte par les approches actuelles. Les systèmes d'annotation efficaces recourent généralement à un prédicteur neuronal d'annotation préalablement entraîné à partir de données issues d'outils classiques. Ce prédicteur demeure ensuite figé tout au long du processus

d'annotation. Cette approche est sous-optimale pour deux raisons principales: (1) le prédicteur entraîné doit déjà être disponible, (2) le prédicteur ne bénéficie pas des nouvelles annotations qui apparaissent au fur et à mesure de l'avancement du processus. Ainsi, un important volume de données annotées est requis dès le départ pour obtenir un outil d'annotation performant, ce qui constitue un défi majeur pour les tâches où les données annotées disponibles sont limitées. Un exemple marquant est la segmentation en IRM du pelvis féminin (Female Pelvis MRI, FPMRI), qui représente l'une des applications clés de cette thèse. En pratique, cela crée une dépendance circulaire, dans laquelle il faut déjà disposer de données annotées pour en générer davantage — un problème qui n'est pas directement traité dans la littérature.

Dans cette thèse, nous répondons aux deux principaux défis liés à l'annotation d'images médicales en proposant quatre contributions distinctes, appliquées à la segmentation de manière spécifique et à l'annotation de façon générale. Ces contributions se répartissent en deux approches centrées sur les données et deux approches axées sur les applications, les premières établissant le socle des secondes. Côté données, nous avons introduit un nouvel ensemble de données de segmentation FPMRI (Female Pelvis MRI dataset, FPMRId) et étudié sa variabilité inter-observateur. Côté applications, nous avons proposé deux solutions majeures répondant aux besoins cliniques et industriels. La première aborde la nécessité d'outils d'annotation adaptés au contexte médical, et la seconde met à disposition un système dédié à l'annotation à grande échelle. Premièrement, nous proposons un cadre général pour la segmentation interactive, non restreint à un domaine spécifique, qui améliore les performances en tenant compte de la manière dont l'utilisateur aborde habituellement la segmentation. En effet, les méthodes existantes de segmentation interactive ne prennent pas en considération l'ordre séquentiel des interactions de l'utilisateur, passant généralement outre la succession des corrections. La solution proposée s'attaque à cette limite en introduisant une « mémoire d'interaction », qui préserve l'ordre des interactions de l'utilisateur et l'intègre dans l'entraînement. Ainsi, le modèle interprète chaque correction au regard des interactions précédentes, ce qui se traduit par une plus grande précision avec moins d'étapes d'interaction. Deuxièmement, nous présentons Single Active Interactive Model (SAIM), un cadre intégrant la sélection de données, l'annotation et l'entraînement au sein d'une architecture unifiée, via un mécanisme de partage de modèle, pour permettre une annotation efficace à grande échelle. SAIM constitue une solution « deux-en-un », dans laquelle le modèle partagé assure simultanément les fonctions d'outil d'annotation et de prédicteur, prêt à être déployé. SAIM fonctionne en boucle : le modèle partagé, pré-entraîné sur quelques centaines d'images, est utilisé pour sélectionner les données les plus informatives à annoter, proposer des annotations et, si besoin, les corriger de manière interactive. Il est ensuite mis à jour à partir de ces annotations nouvellement produites. Ce processus se répète jusqu'à ce qu'il soit arrêté ou que l'ensemble des données disponibles soit annoté. Cette approche atténue le besoin de disposer dès le départ de vastes volumes de données annotées. Concrètement, elle permet de démarrer avec un nombre limité de données annotées, d'améliorer progressivement le modèle et de réduire la charge d'annotation grâce à la sélection de données. Par ailleurs, le modèle reste utilisable pour la tâche visée à chaque étape. Bien que SAIM soit appliqué à la segmentation en imagerie médicale, il n'est en aucun cas limité à une tâche ou un domaine particulier.

Résumé étendu

Contexte. Parmi les tâches fondamentales de l'analyse de données figure la segmentation d'image, qui consiste à attribuer une étiquette à chaque pixel afin de délimiter des structures pertinentes. En imagerie médicale, la segmentation vise généralement à isoler des régions anatomiques ou des lésions et constitue une étape indispensable pour de nombreuses applications. Les segmentations obtenues sont ensuite exploitées pour le diagnostic ou pour divers traitements en aval, tels que la génération de modèles 3D destinés à la réalité augmentée (RA) chirurgicale, comme dans le présent travail. L'une de ces applications est l'assistance par RA lors de la chirurgie coelioscopique de l'utérus. La chirurgie coelioscopique est une procédure mini-invasive réalisée au moyen de petites incisions dans la paroi abdominale, permettant une récupération plus rapide qu'après une chirurgie ouverte. Durant l'intervention, le chirurgien s'appuie sur un écran diffusant en temps réel la vidéo d'un cœlioscope introduit par l'une de ces incisions. Cependant, contrairement à la chirurgie ouverte, où le chirurgien peut voir et palper directement les tissus, la cœlioscopie ne fournit ni retour haptique ni visibilité directe, ce qui complique la procédure. La RA peut pallier ces limitations en offrant des repères visuels supplémentaires et en superposant des informations anatomiques pertinentes dans le champ de vision du praticien. Dans un pipeline RA à l'état de l'art pour la chirurgie cœlioscopique de l'utérus, une imagerie par résonance magnétique (IRM) ou une tomodensitométrie (TDM) préopératoire de l'utérus de la patiente est d'abord segmentée afin de produire un modèle 3D de l'organe et de ses structures internes d'intérêt. Ce modèle 3D est ensuite superposé à l'organe réel dans la vidéo cœlioscopique affichée à l'écran du chirurgien et suivi en temps réel. Le chirurgien peut ainsi « voir à travers » l'organe et planifier les interventions avec une précision accrue.

La production d'un tel modèle en milieu clinique n'est pas triviale. La segmentation doit être précise, générée rapidement et supervisée par des experts médicaux. Même des erreurs mineures peuvent entraîner de graves risques peropératoires ; inversement, un temps d'annotation excessif peut rendre l'ensemble de la solution inapplicable. Satisfaire ces exigences s'avère complexe pour plusieurs raisons. Contrairement aux images naturelles, qui sont abondantes, interprétables et largement étudiées, bénéficiant de ce fait d'outils d'annotation à l'état de l'art, les images médicales présentent des difficultés supplémentaires. Elles sont difficiles à obtenir en raison de réglementations strictes sur la confidentialité, nécessitent une expertise disciplinaire, sont généralement annotées manuellement et demeurent ardues à interpréter. Plus précisément, les techniques d'imagerie médicale telles que la TDM et l'IRM produisent des données tridimensionnelles, obligeant les spécialistes à consacrer un temps considérable à l'examen minutieux des structures volumiques sur plusieurs plans. De surcroît, les régions d'intérêt et les pathologies sont complexes, variables et se prêtent rarement à des représentations uniformes et nettes, ce qui engendre une interprétation subjective : deux experts peuvent diverger quant aux limites et à la nature d'une même région anatomique. Ces contraintes sont particulièrement marquées dans certains domaines spécifiques, comme par exemple l'IRM pelvienne féminine (IRMPF), qui demeurent de ce fait relativement peu explorés. Alors que des domaines de recherche plus explorés, tels que le cerveau ou les poumons, bénéficient d'études plus matures et de modèles de fondation, l'IRMPF ne disposait encore, en 2024, que d'un seul ensemble de données public. Dans ce contexte, la segmentation se heurte à d'importants obstacles : elle est bien plus laborieuse,

chronophage, dépendante de données annotées rares et s'adapte mal à la croissance des jeux de données. Dès lors, une solution idéale pour l'annotation d'images médicales devrait satisfaire quatre exigences principales : (1) être efficace, en réduisant le temps d'annotation tout en améliorant les performances ; (2) être pilotée par un expert avec un effort minimal ; (3) reposer sur un jeu de données réaliste, effectivement obtenable et annotable en pratique ; et (4) être évolutive, afin de rendre faisable l'annotation de vastes jeux de données d'imagerie médicale. Conformément à ces critères, les approches d'apprentissage automatique (Machine Learning, ML) ont démontré une performance supérieure pour de nombreuses tâches médicales.

L'apprentissage automatique est un domaine d'étude qui se concentre sur le développement d'algorithmes statistiques permettant aux systèmes d'apprendre à partir de données et de faire des prédictions ou de prendre des décisions sans recourir à des instructions explicitement programmées. Ces algorithmes se concrétisent sous forme de modèles, lesquels représentent les motifs et les relations appris au sein des données. Le processus d'apprentissage, appelé « entraînement », consiste à ajuster les paramètres du modèle en fonction des données d'entrée. Une fois entraîné, le modèle peut être employé pour effectuer des prédictions ou prendre des décisions sur de nouvelles données, encore jamais observées. L'annotation des données constitue une pierre angulaire de l'apprentissage automatique : elle consiste à associer, à chaque point de données, des étiquettes vérités terrain, des catégories descriptives ou des marqueurs. Ces annotations orientent le processus d'entraînement et aident à garantir que le modèle assimile des informations pertinentes, aboutissant à de meilleures prédictions. Un modèle d'apprentissage automatique est fréquemment un réseau de neurones artificiels, formé de couches de neurones artificiels interconnectés. L'apprentissage profond (Deep Learning, DL) est une sous-catégorie de l'apprentissage automatique qui exploite des réseaux dits « profonds », caractérisés par un nombre accru de couches, offrant souvent de meilleures performances, mais nécessitant en contrepartie un volume de données d'entraînement plus important.

Face aux défis exposés, cette thèse traite l'ensemble des composantes majeures d'une solution d'apprentissage automatique pour la segmentation d'images médicales : collecte, annotation et analyse des données ; conception du système de segmentation ; conception d'un cadre de montée en charge de l'annotation. Plus précisément, nous présentons quatre contributions distinctes, regroupées en deux approches centrées sur les données et deux approches centrées sur l'application, les premières servant de fondement aux secondes. Le premier groupe, axé sur les données, comprend : (1) le jeu de données IRMPF et (2) l'étude de variabilité inter-experts menée sur ce jeu. Le second groupe, orienté application, rassemble : (3) un système de segmentation interactive et (4) un cadre économe en données pour l'annotation à grande échelle appliquée à la segmentation. Nous détaillons chacune de ces quatre contributions dans les sections suivantes.

Jeu de données IRMPF. Nous avons constitué et annoté un jeu de données de segmentation IRMPF avec l'appui d'experts médicaux du CHU de Clermont-Ferrand. À notre connaissance, il s'agit du premier jeu de cette ampleur comportant les segmentations des structures anatomiques suivantes : utérus, vessie, cavité utérine, col de l'utérus, fundus et paroi antérieure ; ainsi que les annotations des pathologies suivantes : tumeurs, endométriose et adénomyose. Ce jeu de données reflète la complexité de la segmentation IRMPF et présente des annotations multi-classes, multi-étiquettes, multi-instances et multi-composants. Il se compose de volumes IRM annotés manuellement par des radiologues experts, couvrant de fortes variations de forme, taille et texture

de ces structures. Notre contribution porte à la fois sur le jeu de données lui-même et sur son processus de création : collecte des données, lignes directrices d'annotation et défis associés. Les annotations ont été réalisées sur des plateformes spécialisées, et le développement du corpus a été suivi, documentant l'évolution de la collecte, de l'annotation et l'impact des changements de plateforme. Le jeu de données est désormais stabilisé ; il a servi à l'entraînement et à l'évaluation des autres contributions, à l'étude utilisateur de la contribution #3 et à l'analyse de variabilité inter-experts de la contribution #2.

Étude de variabilité inter-experts. Nous avons conduit, en collaboration avec un expert médical, une étude mono-centrique de variabilité inter-experts. Plus précisément, nous avons réalisé une analyse rétrospective portant sur 10 patientes ayant bénéficié d'une IRM pelvienne 1,5 T avec séquences axiales T2 Propeller de 5 mm d'épaisseur. Les volumes ont été segmentés par 6 radiologues d'expérience variable, générant des segmentations de l'utérus, de la vessie, des myomes utérins, de la cavité utérine et du col de l'utérus. Pour quantifier la corrélation entre experts, nous avons calculé à partir de ces segmentations : (1) le coefficient de Dice, comparé deux à deux entre experts, et (2) le volume de chaque structure anatomique pour chaque expert. Nous avons ensuite agrégé ces résultats en calculant, pour chaque expert puis pour chaque série, la moyenne et l'écart-type de ces métriques. Par ailleurs, l'algorithme STAPLE a été utilisé afin de générer, pour chaque patiente, une segmentation de référence produite à partir des segmentations de l'ensemble des radiologues ; la corrélation de chaque segmentation individuelle à cette référence a également été évaluée. L'étude a mis en évidence : une excellente corrélation inter-experts pour le volume utérin, une très bonne corrélation pour les fibromes et la vessie, une corrélation satisfaisante pour la cavité utérine et une corrélation modérée pour le col de l'utérus. Ces résultats indiquent que la segmentation de l'utérus est un processus fiable et reproductible, appuyant le développement potentiel d'outils de segmentation automatiques ou semi-automatiques.

Segmentation neuronale interactive. La segmentation d'images médicales peut être réalisée manuellement, automatiquement ou à l'aide d'algorithmes semi-automatiques (interactifs). Compte tenu des enjeux élevés des applications médicales, les algorithmes interactifs sont les plus adaptés au milieu clinique : l'expert peut corriger, si nécessaire, la segmentation proposée par le modèle. Pour que cette correction soit efficace, il faut que le modèle interprète les corrections de l'utilisateur et en déduise son intention, point que les approches existantes ne considèrent pas complètement. En particulier, les méthodes interactives actuelles n'exploitent pas l'ordre séquentiel des interactions, négligeant la chronologie des corrections. La solution proposée surmonte cette limite grâce à une mémoire d'interaction qui préserve l'ordre des entrées utilisateur et l'intègre à l'entraînement.

Nous présentons un système interactif général de segmentation d'images multi-classe fondé sur l'apprentissage profond, dans lequel un réseau de base est placé dans une boucle d'interaction utilisateur dotée d'une mémoire des interactions. Cette mémoire est modélisée explicitement comme une séquence d'états successifs du système, à partir de laquelle le réseau apprend ses représentations, assimilant ainsi le processus de raffinement de la segmentation. L'entraînement est complexe, l'entrée du réseau dépendant de sa sortie précédente. Nous adaptons le réseau à cette boucle en introduisant un utilisateur virtuel, modélisé par la simulation dynamique des clics itératifs de l'utilisateur réel. Nous avons évalué notre système face aux méthodes existantes

sur des tâches de segmentation multi-classe exigeantes, notamment l'IRMPF ainsi que la segmentation du foie et du pancréas sur scanner, en utilisant des jeux de données internes et publics. Une étude utilisateur menée auprès de onze professionnels de santé a montré une réduction significative du temps d'annotation avec notre système par rapport aux outils traditionnels. Nous avons analysé systématiquement l'influence du nombre de clics sur la précision : après un seul cycle d'interaction, notre système surpasse les solutions automatiques de configuration comparable. Nous présentons une étude d'ablation et montrons que notre cadre surpasse les systèmes interactifs existants.

Annotation interactive avec peu de données et entraînement simultané du modèle. Les systèmes d'annotation efficaces reposent généralement sur un prédicteur neuronal, d'abord entraîné sur des données issues d'outils classiques, puis figé pendant tout le processus d'annotation. Cette approche présente deux limites majeures : (1) le prédicteur doit déjà être disponible, et (2) il ne bénéficie pas des nouvelles annotations produites au fil du temps. Ainsi, un volume important de données annotées est requis dès le départ pour disposer d'un outil performant, ce qui est particulièrement problématique lorsque ces données sont limitées. C'est le cas, par exemple, de la segmentation IRMPF, l'une des applications clés de cette thèse. Il en résulte une dépendance circulaire : il faut des données annotées pour produire davantage de données annotées, un problème peu abordé dans la littérature.

Nous proposons un cadre appelé SAIM, qui intègre en une seule architecture les trois étapes de sélection des données, d'annotation et d'entraînement. SAIM se distingue des travaux existants par trois propriétés : (1) il utilise un prédicteur interactif profond ; les outils classiques ne sont donc pas nécessaires et le prédicteur peut être pré-entraîné avec peu de données pour fournir des annotations de qualité ; (2) un modèle unique est partagé entre les trois étapes, de sorte qu'il reste déployable et s'améliore au fur et à mesure des annotations ; (3) SAIM recourt à l'apprentissage actif pour maximiser l'impact de chaque annotation sur les performances du prédicteur, accélérant ainsi sa progression. En pratique, SAIM est une solution « deux-en-un » : le modèle partagé sert à la fois d'outil d'annotation et de prédicteur prêt à l'emploi. SAIM fonctionne en boucle : pré-entraîné sur quelques centaines d'images, le modèle sélectionne les données les plus informatives, propose les annotations et les corrige de manière interactive si besoin, puis il est mis à jour avec ces nouvelles annotations. Le processus se répète jusqu'à l'arrêt ou jusqu'à l'annotation complète du corpus. Cette approche réduit le besoin initial de vastes jeux de données annotées ; elle permet de démarrer avec un volume restreint, d'améliorer progressivement le modèle tout en diminuant la charge d'annotation grâce à la sélection de données, et de maintenir le modèle déployable à chaque étape.

Nous avons évalué SAIM dans des scénarios d'annotation simulés en mode automatisé, sur des jeux de données de segmentation intégralement annotés couvrant cinq tâches : (1) segmentation sémantique multi-classe IRM du pelvis féminin ; (2) segmentation sémantique multi-classe du foie sur scanner ; (3) segmentation sémantique multi-classe du pancréas sur scanner ; (4) segmentation cardiaque IRM, où SAIM est confronté à huit approches d'apprentissage semi-supervisé (semi-supervised learning, SSL) parmi les meilleures de l'état de l'art ; (5) segmentation d'images naturelles, où SAIM est comparé à trois approches d'auto-apprentissage (self-training, ST) également à l'état de l'art. Nous avons en outre démontré SAIM dans un scénario réel d'annotation de segmentation rénale IRM avec un utilisateur humain et estimé le gain

de temps par rapport aux outils classiques. SAIM surpasse aussi bien les outils traditionnels que les approches de l'état de l'art. Bien qu'appliqué ici à la segmentation d'imagerie médicale, SAIM n'est lié à aucun domaine spécifique : il permet de lancer une annotation interactive à grande échelle à partir d'un jeu de données limité et de minimiser la quantité de données à annoter, tout en améliorant itérativement les performances.

Valorisation

Workshop publications:

- **A Deep Learning-based Interactive Medical Image Segmentation Framework**
  Mikhailov, I., Chauveau, B., Bourdel, N., Bartoli, A.
  AMAI - The First Workshop on Applications of Medical AI at MICCAI, 2022
  Best student paper award, honorary mention

- **Sharing is Caring: Concurrent Interactive Segmentation and Model Training using a Joint Model**
  Mikhailov, I., Chauveau, B., Bourdel, N., Bartoli, A.
  CVAMD - Workshop on Computer Vision for Automated Medical Diagnosis at ICCV, 2023

Journal publications:

- **A Deep Learning-based Interactive Medical Image Segmentation Framework with Sequential Memory**
  Mikhailov, I., Chauveau, B., Bourdel, N., Bartoli, A.
  Computer Methods and Programs in Biomedicine, 245, Mar. 2024.

- **Sharing is Caring: Concurrent Interactive Segmentation and Training using a Shared Model**
  Mikhailov, I., Chauveau, B., Bourdel, N., Bartoli, A.
  Submitted to *Computerized Medical Imaging and Graphics*, 2025.

Patent applications:

- Système et procédé de segmentation d'image semi-automatique par apprentissage à boucle d'interaction utilisateur et procédé d'entraînement associé (Mikhailov and Bartoli, 2024b)
  - **France:** Submitted on 28 Feb. 2022, delivered on 19 Jul. 2024.
  - **PCT Application:** Submitted on 28 Feb. 2023
    - **Canada:** Submitted on 13 Aug. 2024, under review
    - **Japan:** Submitted on 23 Aug. 2024, under review
    - **China:** Submitted on 28 Aug. 2024, under review
    - **USA:** Submitted on 28 Aug. 2024, under review
    - **India:** Submitted on 28 Aug. 2024, under review
    - **EU:** Submitted on 13 Sep. 2024, under review

- Système et procédé combiné de sélection, d'annotation et d'entraînement via un modèle d'apprentissage automatique partagé (Mikhailov and Bartoli, 2024a)

– **France:** Submitted on 9 Mar. 2023, under review

– **PCT Application:** Submitted on 8 Mar. 2024, under review

Acknowledgements

I would like to begin by expressing my deepest gratitude to everyone who helped me along this PhD journey.

First, my supervisor **Adrien Bartoli**, for his strong guidance, for being deeply involved, for making himself available, and for always setting a high bar. Both his scientific advice and strategic planning were greatly beneficial for my work. I feel that I learned a lot, and not only in my discipline, but in research on a professional and holistic level.

I am grateful to **SURGAR**, its founding members, collaborators, and the entire team, for this opportunity, for welcoming me from its inception, and for cultivating a motivating and energizing atmosphere throughout the duration of my CIFRE thesis. Being able to contribute at the intersection of industry and research is something I truly enjoy and look forward to continuing.

My heartfelt thanks go to all of my colleagues at SURGAR and in the laboratory: **Kilian C.**, **Julie D.**, **Richard M.**, **Khadija H.**, **Bao B.** and others, past and present. Over these years we have shared countless hours of work, discussions, and more than a few memorable holidays together. Your camaraderie is what I will remember.

To my family—my father **Sergey**, my brother **Fedor**, and my mother **Elena**—thank you for a level of support that simply cannot be quantified. I also wish to honour the memory of my late grandmother **Ludmila** and grandfather **Viktor**, who not only valued but contributed to the advancement of science, and to extend my deep gratitude to my grandmother **Elvira**, my aunt **Irina**, and my uncle **Viktor** for their encouragement throughout this journey.

I am also fortunate to count on wonderful friends in both France and Russia: **Dmitry S.**, **Rita F.**, **Sergei D.**, **Anastasia E.**, **Anzhelika F.**, and **Roman S.** and others. Whether we were able to meet in person or keep in touch online, your friendship and our time spent together has been a steady source of motivation and joy.

Finally, to my partner **Elena**: thank you for your love, understanding, and unwavering support. With you I always feel renewed and ready for what comes next.

To everyone mentioned here—and to those whose names may not appear but whose contributions were nonetheless significant: please accept my deepest appreciation. Done!

Contents

Chapter 1

Introduction

1.1 Machine Learning

1.1.1 General Points

ML is a discipline within the domain of Artificial Intelligence (AI) (Zhang and Lu, 2021). The goal of AI is to automate the process of constructing models or systems to perform tasks that traditionally required human intelligence, making it possible for machines to understand and interpret data with a high degree of accuracy. At its core, AI involves techniques that enable computers to mimic human behaviour, reproducing or often outperforming human decision-making in tackling complex problems, either autonomously or with minimal human input. This field addresses numerous challenges, such as knowledge representation, reasoning, learning, planning, perception, communication, and others, utilising a diverse array of tools and methods.

Specifically, ML deals with the creation and study of statistical algorithms, which can learn from existing data and generalise to new, unseen data, thereby enabling tasks to be performed without explicit programmed instructions. In contrast to using an explicit handcrafted set of rules to construct an analytical model, ML algorithms seek to learn meaningful relationships and patterns from examples and observations to then make predictions or decisions. Such algorithms usually improve their performance iteratively by incorporating the data they are exposed to through a process called training. Then, they serve as analytical models, which input data and output the desired inferred information.

The process of training an ML model involves adjusting the model's internal parameters based on the data it is being continuously fed. Such data is called training data, as opposed to validation and evaluation data, which are used to estimate the model's performance. During training, the model is exposed to numerous examples with known outcomes within the training dataset, and gradually learns to replicate these outcomes by minimising errors in its predictions on validation and evaluation datasets.

An example of an ML model is an ANN. ANN is a computational model inspired by the brain, which simulates the way biological neurons process information (Rosenblatt, 1958). An ANN consists of mathematical models of interconnected processing units known as artificial neurons. The most simple form of an ANN is the basic perceptron. It consists of one layer of input nodes connected to an output node, making it a single-layer ANN only suitable for solving linear problems. A schematic of the perceptron is shown in figure 1.2. More advanced ANNs expand on this by incorporating additional layers and constructing more complex architectures, allowing them to handle

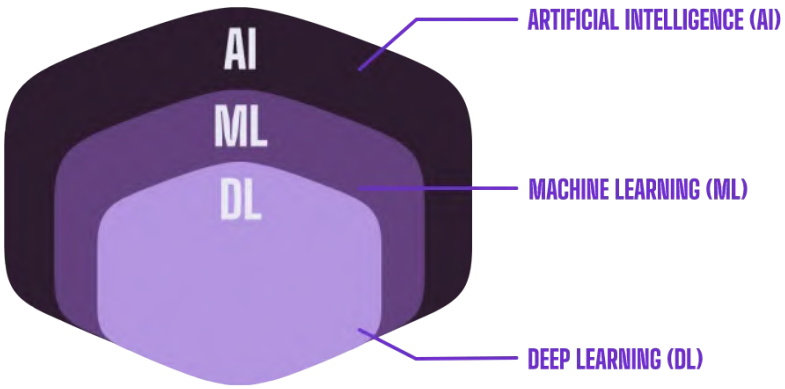


Figure 1.1: Visualization of the hierarchical relationship among the domains of AI, ML and DL, highlighting their nested structure. Image source: (Cooler Master, 2024)

non-linear and generally more complex tasks.

ML has gained widespread applicability in recent years, achieving good performance in various fields such as computer vision, speech recognition, natural language processing and others. This progress is largely due to the development of ANNs into more complex deep neural network architectures with many hidden layers, which have significantly enhanced learning capabilities. This advancement, known as DL (Talaei Khoei et al., 2023), has enabled systems to achieve remarkable performance levels, demonstrating superhuman performance in speech, image and handwriting recognition, reading comprehension, language understanding and other tasks in many domains (Kiela et al., 2021, 2023). The hierarchical relationship between the domains of AI, ML and DL is illustrated in in figure 1.1.

While DL is considered to be a subtype of ML, it is often contrasted with traditional 'shallow' ML. This is due to two key specifics of the latter: (1) it often requires feature engineering done by a domain specialist to extract data representations and (2) it often shows poor performance on complex data. The shift from ML to DL can thus be described as moving from specialised feature engineering to general feature learning. In traditional ML, domain expertise is crucial to identify and design relevant features. This means that despite traditional ML's ability to model complex problems, its performance is often constrained by the quality of handcrafted features. DL, on the other hand, replaces these handcrafted features with generic, adaptable features that are automatically learned during the training process. This is achieved by increasing the depth of a neural network via the number of layers. Specifically, compared to the basic ANN, Deep Artificial Neural Network (DNN) generally comprises multiple hidden layers, structured in nested architectures and feature more sophisticated neurons (Janiesch et al., 2021). The difference between a simple ANN, which can be also called a Shallow Artificial Neural Network (SNN), and a DNN is shown in figure 1.4. A key advantage of DL over traditional ML is its superior performance in handling large datasets and its ability to process unstructured data, such as images and natural languages. This makes DL particularly effective in scenarios where a high level of abstraction is needed to identify complex patterns and relationships between data points. Figure 1.3 compares how performance scales with increasing data for DNN, SNN, and traditional ML. At the same time, DL requires sub-

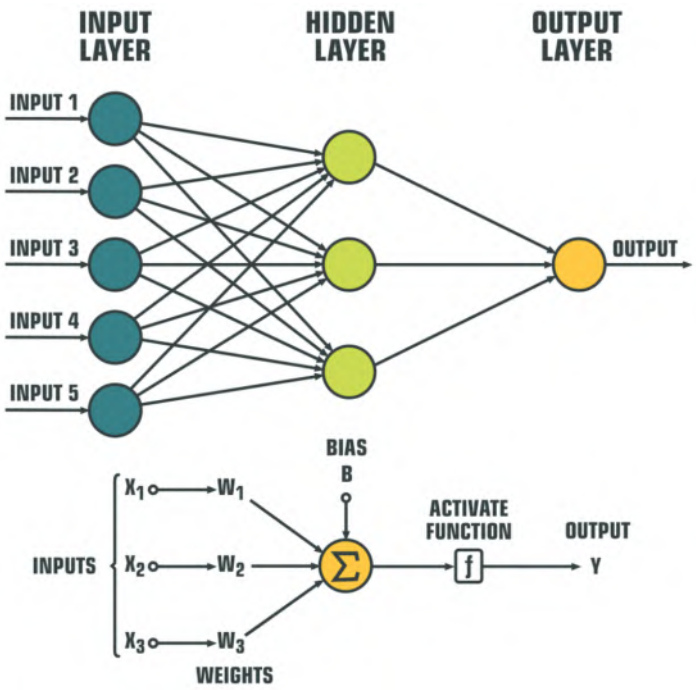


Figure 1.2: A schematic of perceptron. Top: general schematic with input, hidden, and output layers. Bottom: in-detail equivalent of the general schematic showing weighted inputs, bias, summation, and activation function. Image source: (Seidel et al., 2020).

stantial amounts of annotated training data to achieve high accuracy, which is difficult to obtain in domains where data availability is limited due to legal, technical, ethical and other concerns. For instance, in the medical field, obtaining annotated data can be challenging due to the rarity of the medical condition in question and due to the patient records containing sensitive information.

1.1.2 Tasks

Overview

The main ML tasks can be generally aggregated into the following 7 categories (Sarker, 2021b; Alzubaidi et al., 2021; Shinde and Shah, 2018; Barragán-Montero et al., 2021): (1) classification, (2) regression, (3) clustering, (4) dimensionality reduction, (5) detection, (6) registration and (7) segmentation.

Classification. Classification is the task of assigning input data into one of several predefined categories or labels. More precisely, classification is essentially a function f that maps input variables X to output variables Y, with Y representing the target categories or labels (Han et al., 2022). A basic example is email classification, where emails are sorted into categories 'spam' or 'not spam'.

Regression. Regression involves predicting a continuous numerical value based on input data. It uses various methods to model the relationship between one or more predictor variables X and a continuous outcome variable Y (Han et al., 2022). A basic example is predicting house prices based on features like location, size, and the number of bedrooms. The key difference between classification and regression is that the former predicts discrete class labels, while the

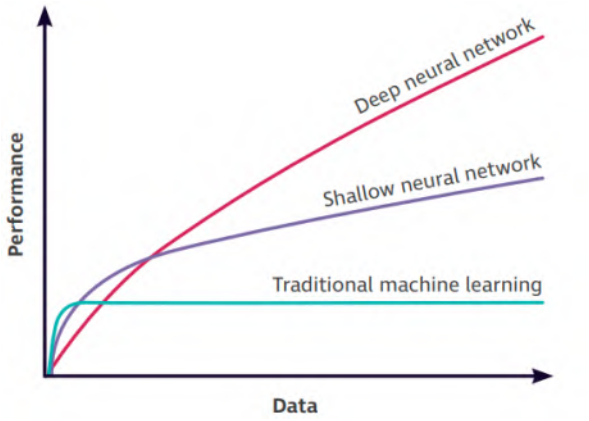


Figure 1.3: Practical performance scaling with increase in data for DNN, SNN, and traditional ML. The image is adapted from (Ng, 2016).

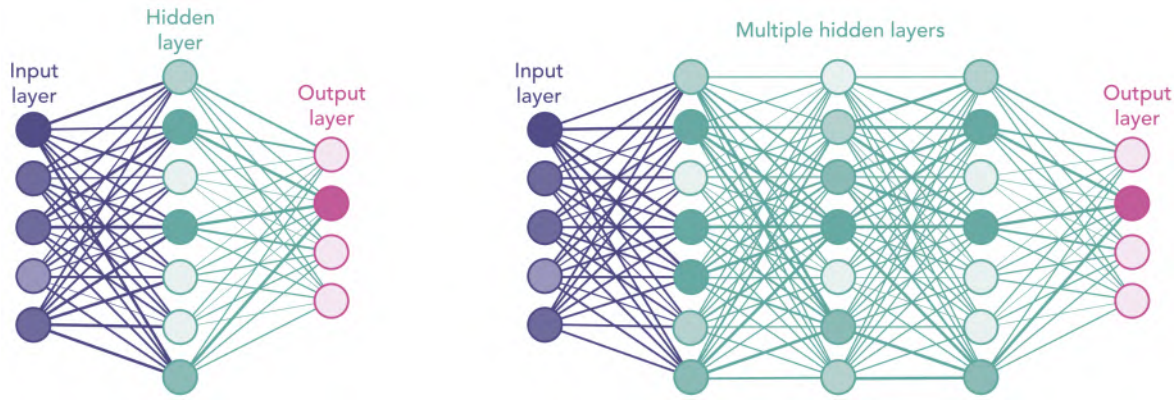
latter predicts continuous quantities. Regression models are extensively applied across numerous domains, including financial forecasting, where they predict market trends and asset prices; cost estimation for project planning; trend analysis in marketing; time series estimation for understanding temporal data patterns; and modelling drug responses in pharmacology.

Clustering. Clustering is the process of grouping a set of objects or data points by similarity. Therefore, the objects in the same category (or cluster) are more related to each other than to those in other groups (Han et al., 2022). It is often used as a data analysis technique to discover trends or patterns in data. A basic example is customer segmentation, where customers are grouped based on purchasing behaviour.

Dimensionality reduction. ML data processing is challenging due to the abundance of high-dimensional data. Thus, dimensionality reduction targets reducing the number of random variables under consideration by obtaining a set of principal variables. For this purpose, both feature selection and feature extraction are used. Feature selection involves retaining a smaller subset of the original features (Sarker et al., 2020a), while feature extraction creates new features from the existing data (Sarker et al., 2020b). A basic example of a dimensionality reduction method is Principal Component Analysis (PCA) (Pearson, 1901), which reduces the dimensions of data while preserving as much variance as possible.

Detection. Detection refers to identifying the presence and location of objects within a given context, often involving both classification and localization (Kaur and Singh, 2023). Classification assigns input data to predefined categories, determining what an object is. Localization specifies the precise position of these objects, typically by providing bounding boxes around them. Thus, detection identifies not only the class of objects, but also whether they are present in an image or dataset and where. It is commonly used in computer vision with a basic example being object detection in images, such as detecting and locating cars and pedestrians in a street photo or video.

Registration. Registration is the process of aligning data from different sources into a single coordinate system (Unberath et al., 2021). The input data for registration can take various forms, such as point clouds, voxel grids, and meshes. Registration is often used in medical imaging and



Simple Artificial Neural Network (ANN) **Deep Artificial Neural Network (DNN)**

Figure 1.4: The basic difference between the simple Artificial Neural Network (left) and Deep Artificial Neural Network (right) architectures.

Image source: (Waldrop, 2019).

computer vision to align data from multiple sources or from different viewpoints. A common example is aligning Positron Emission Tomography (PET) and Magnetic Resonance Imaging (MRI) scans of the same patient to provide a comprehensive view that combines metabolic information from PET with anatomical details from MRI. Another example is aligning 3D scans of an object taken from multiple perspectives to create a complete 3D model.

Segmentation. Segmentation involves partitioning data into distinct segments for further analysis. It can be seen as the classification of each data point: for example, assigning a label to every pixel in an image or to every word in a text. This task is commonly used in fields such as image processing and natural language processing.

Image Segmentation

Image segmentation is a crucial component of many visual understanding systems. It involves dividing images into distinct segments or objects (Szeliski, 2022). Specifically, image segmentation entails classifying each pixel (in 2D) or voxel (in 3D) into a class that represents the object it belongs to. As opposed to image classification, which assigns a single label to the entire image, image segmentation provides detailed understanding at the pixel or voxel level and is generally more complex. Examples of image segmentation results are shown in figure 1.5.

There are three general types of image segmentation, grouped based on the envisioned goal (Minaee et al., 2021). They are semantic segmentation, instance segmentation, and panoptic segmentation:

- **Semantic segmentation** involves labelling each pixel in an image with a category from a predefined set of object classes, such as human, car, tree, or sky.

- **Instance segmentation** extends semantic segmentation by identifying and delineating each distinct object instance within the image. However, it specifically focuses on instances rather than the entire scene. For example, in an image of a group of people, instance segmentation would partition each individual person, assigning a unique identifier to each one, while ignoring non-object regions like the background.

Figure 1.5: Examples of image segmentation results on natural scenes. Object segmentations are represented with semi-transparent coloured masks directly overlaid on the objects themselves. All images show instance segmentation except the one in the bottom row's centre, which is an example of semantic segmentation. Image sources: (Cheng et al., 2021; Cordts et al., 2016).

- **Panoptic segmentation** combines the strengths of both semantic and instance segmentation. It provides a comprehensive view by labelling each pixel with both the object category and distinguishing between instances where applicable. This approach offers a unified framework that handles both stuff (amorphous regions like sky or grass) and things (distinct objects like cars or people).

The literature features a wide array of image segmentation algorithms. Among the early techniques are thresholding (Otsu et al., 1975), region growing (Nock and Nielsen, 2004), k-means clustering (Dhanachandra et al., 2015), and watershed algorithms (Najman and Schmitt, 1994). Other methods incorporate advanced mathematical models and optimization techniques to handle complex image structures. Among them are active contours (Kass et al., 1988), graph cuts (Boykov and Jolly, 2001) conditional random fields and Markov random fields (Plath et al., 2009), and sparsity-based approaches (Starck et al., 2005). In recent years, however, DL became a dominant force in image segmentation in many tasks, with preceding methods now mostly considered classical. DL offers significant performance enhancements and often achieves top accuracy scores on widely-used benchmarks. Notable examples of such DL methods are SAM (Segment Anything Model) (Kirillov et al., 2023) and SAM 2 (Ravi et al., 2024) targeting task-agnostic image segmentation, where the latter shows a significant improvement over the State of the Art (SOTA) methods for both image and video segmentation (Ravi et al., 2024).

Image segmentation plays a crucial role in various domains. Some of the notable applications are the domains of medical image analysis, autonomous driving and remote sensing:

- In the field of **medical imaging**, advanced segmentation techniques, particularly those based on DL, assist radiologists by improving the accuracy and efficiency of interpreting medical scans (Tajbakhsh et al., 2020). For example, image segmentation is pivotal for tasks such as tumor detection and disease diagnosis, such as in (Amyar et al., 2020) for CT classification and segmentation to simplify COVID-19 screening.

- In **autonomous driving**, image segmentation is essential for enabling vehicles to perceive and understand the environment. By segmenting visual input into distinct categories, such

as roads, pedestrians, vehicles, and obstacles, autonomous driving systems can make informed decisions about navigation and collision avoidance (Muhammad et al., 2022).

- **Remote sensing** concerns the analysis of satellite images. Image segmentation allows researchers to track changes over time, monitor natural resources, and plan sustainable development. Specifically, some of the applications are land cover classification, environmental monitoring, precision agriculture (Paoletti et al., 2019) and urban planning (Gao et al., 2020).

1.1.3 Applications

ML in general and DL specifically are widely applied across various sectors, including medicine, natural language processing, speech recognition, cybersecurity, smart agriculture, business and financial services, virtual assistant and chatbot services, object detection and recognition, recommendation and intelligent systems (Sarker, 2021a) and others. The diversity of tasks being addressed with deep learning technologies is very high, including diagnosing COVID-19 (Islam et al., 2020), cancer classification (Sevakula et al., 2018), sentiment analysis (Wang et al., 2019b), object detection in X-ray images (Gu et al., 2020), network intrusion detection (Al-Qatf et al., 2018), stock trend prediction (Ishwarappa, 2021), and disaster management (Aqib et al., 2018). Statistics on domains of application of notable AI systems is shown in figure 1.6.

Applications of ML in medicine are of special importance, since it presents unique challenges that demand specifically-tailored solutions. The medical domain is characterised by time-sensitive decision-making, high stakes involving patient health and safety, and the requirement for experts to be constantly responsible for making numerous critical clinical decisions at all times. ML addresses these challenges in 3 main ways: (1) by quickly processing vast amounts of complex data, not feasible or doable by human experts (Abadia et al., 2022), (2) by automating routine diagnostic tasks, thus reducing the burden on experts (Fan et al., 2022) and (3) by enhancing precision in diagnosis and treatment, allowing for detection of patterns and anomalies that may be overlooked or difficult to identify (Najjar, 2023).

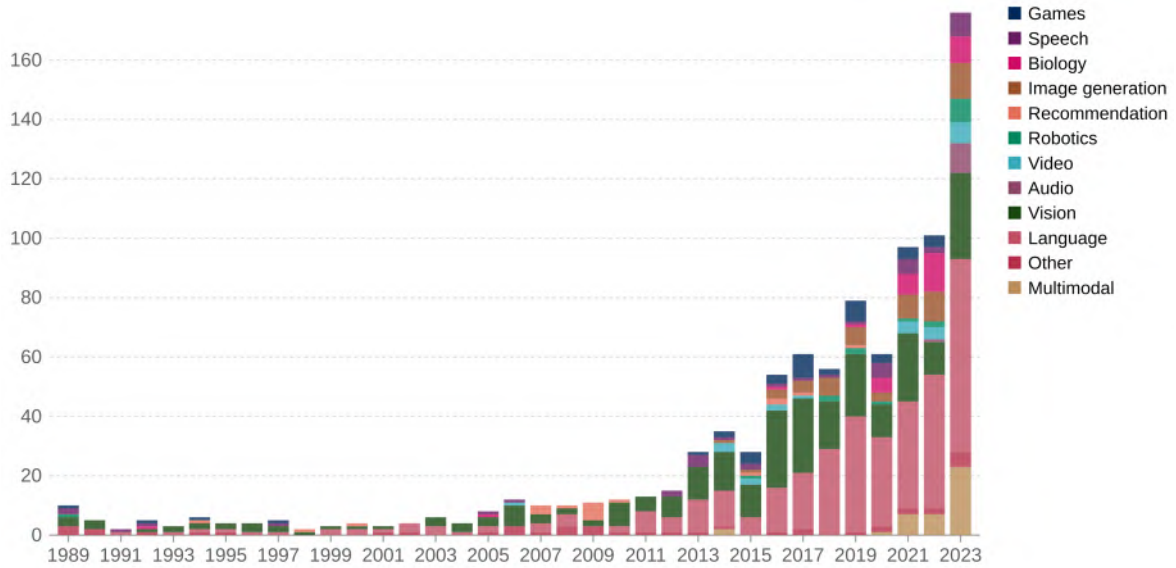


Figure 1.6: General domains of notable AI systems by year of publication, starting from 1989. The authors define 'notable' systems based on multiple criteria, such as advancing the state of the art and having historical significance. Image source: (Giattino et al., 2023).

1.1.4 Challenges

ML approaches in general, and DL in particular, face 6 key challenges. They are: (1) data availability and quality, (2) data annotation, (3) data imbalance, (4) interpretability and explainability, (5) catastrophic forgetting and (6) underspecification.

Data Availability and Quality. Although ML models can perform well with less data, they often require substantial amounts of annotated data to scale effectively and achieve their best potential performance. However, obtaining the data can be expensive and time-consuming, particularly in fields like medicine, where data is often limited and contains sensitive information. Furthermore, the quality of this data can vary significantly, with issues such as incomplete annotations, noise, and inconsistency, which can hinder model performance.

Data Annotation. Data annotation is a crucial step in the development of ML algorithms, as it involves producing accurate labels to ensure that ML algorithms learn meaningful patterns and make reliable predictions. In many fields, including medicine, the annotation process is complex and costly due to the need for expert input and the detailed nature of the data. Medical data, in particular, must be meticulously curated and annotated by experts, such as radiologists or pathologists, to ensure that annotations accurately reflect the conditions depicted in the data. This requirement adds significant complexity and expense to the process, as these experts are in high demand and limited in availability.

Data Imbalance. Dataset imbalance means that certain classes contain significantly more samples than others. This imbalance can pose significant challenges to the development of effective ML models (Johnson and Khoshgoftaar, 2019). For instance, in medicine, datasets often have a skewed ratio of positive to negative cases, with negative cases far outnumbering the positive ones. This is commonly seen in medical imaging datasets where the number of healthy images vastly exceeds the number of images showing pathological conditions. Such imbalance can lead to biased models that perform well on the majority class but poorly on the minority class. This is particularly concerning in medical applications where the accurate detection of minority cases, such as rare diseases or conditions, is crucial for patient care and treatment. Models trained on imbalanced data may fail to detect these minority cases effectively, leading to increased false negatives, where the model incorrectly identifies an unhealthy case as healthy.

Interpretability and Explainability. As ML approaches become more complex, their decision-making processes often resemble 'black boxes', making it difficult to understand how they reach specific predictions. This opacity arises from their architecture: for example, deep neural networks can contain millions of parameters spread across numerous layers. These models learn features from data, but the connections between input data and final predictions are not readily interpretable by humans. This lack of transparency poses significant challenges in high-stakes fields like healthcare and finance, where decision-making transparency is crucial.

Catastrophic Forgetting. Specific to DL, there is a challenge in incorporating new information into a DL model, which arises from the model losing the ability to recall previously learned data (Lee et al., 2019; Wang et al., 2023). For instance, if a model trained to classify 1,000 types of

flowers is fine-tuned with a new flower type, it might perform poorly on the original classes. This issue is common in tasks where continuous flow of data is expected.

Underspecification. Underspecification is a challenge, which arises when models exhibit strong performance during training but falter in real-world applications (D'Amour et al., 2022). This issue occurs because minor alterations in data or environmental conditions can result in significant variations in model predictions, compromising their reliability. To mitigate underspecification, it is crucial to assess how models perform under a wide range of conditions, helping to uncover and address potential points of failure.

1.2 Data Annotation

1.2.1 General Points

Data annotation is the process of assigning specific descriptions, known as labels, to each data point in a dataset. These labels can be directly used for analytical purposes or to produce new data representations. Currently, they often serve as the 'ground truth' information for the ML models, where they are employed for learning, validation and evaluation. As per definition, ground truth is considered unquestionably true, thus labels reflect the contents or desired outputs of a model for each data point.

It can be seen that a large part of the ML challenges are directly dependent on the input data. Currently, the volume of data used for training remains one of the main drivers of the model's performance growth with no end in sight (Liu et al., 2023). Hence, there is a need for large, high-quality datasets to ensure robust model performance. Thus, the key challenge is efficient and accurate data annotation, which is difficult to achieve: it is exacerbated by the need for domain expertise, difficulties in data interpretation and critical nature of model decisions in many domains - for example, in medical imaging (Whang et al., 2023).

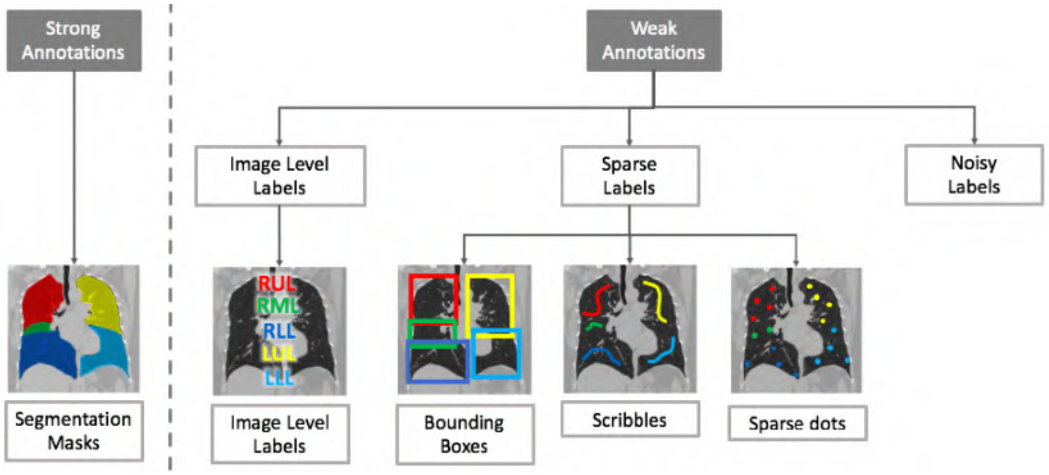


Figure 1.7: Examples of image-level annotations for lung lobes in a coronal view of a chest CT. The annotations are divided into two categories: strong and weak, based on the exhaustiveness and precision of the information provided. Noisy annotations can result from inaccuracies or inconsistencies introduced by any annotation tool or process, which is why an example is not provided. The image is adapted from (Tajbakhsh et al., 2020).

Data annotation applies to all kinds of data, and the annotations can contain any kind of information. However, the exact process of data annotation varies significantly depending on the type of data and the tools employed. This is clearly seen with the following three basic distinctive data types - images, text and audio:

- **Images** are annotated with labels that identify and categorise visual elements within the picture. Image annotation might include drawing bounding boxes around objects, identifying pixel-wise segments, or marking whole images for classification tasks. Some examples of image-level annotations are shown in figure 1.7.

- **Text** is tagged for various attributes such as sentiment or topic categories. Text annotations help in understanding the structure and meaning of text, crucial for applications like natural language processing.

- **Audio** is often labelled with transcriptions, speaker identifications, or emotional tone. Audio annotations enable tasks like speech recognition and sentiment analysis.

1.2.2 Approaches

Traditional data annotation often relies on extensive manual effort, where humans annotate data based on their understanding and expertise mostly using classical tools. An annotation tool is usually a piece of software with a Graphical User Interface (GUI), which allows the annotators to directly interact with the data and provide annotations using a number of instruments. The underlying functionality is task- and data-dependent. For example, in the context of image annotation, these tools enable users to import images and use drawing and tagging instruments to create bounding boxes, polygons, or detailed segmentation masks directly on the images. Many of these tools also integrate some level of automation in order to assist annotators. For example, they might automatically suggest object boundaries based on its colour intensity. The output usually needs to be adjusted due to the imprecision of such methods. This blend of manual and semi-automated targets reduces the annotation time, while benefiting from manual precision.

More advanced data annotation methods often incorporate ML and, more specifically, DL. These methods may reduce the reliance on purely manual annotation directly or indirectly. Direct methods enhance the annotation instruments. Typically, it is a DL model trained on a large dataset, which is employed to suggest annotations then validated or corrected by the annotator. Indirect methods address other aspects of the data annotation process. They include crowdsourcing annotations (Su et al., 2012), data selection in order to annotate only select data points (termed 'active learning' (Tharwat and Schenck, 2023)) and usage of data with no, incomplete or noisy annotations.

Data annotation is especially challenging in the context of image segmentation, which is time-consuming due to per-pixel annotations and requires high precision, since even minor inaccuracies can lead to significantly different outcomes in applications such as medical imaging or autonomous vehicle navigation. To address this, image segmentation is approached from multiple angles in the literature. The main ones are: (1) the degree of user involvement and (2) the nature of the annotation algorithm.

1.3 Image Segmentation by User Involvement

Image segmentation approaches may be split in three categories, depending on the degree of user involvement. They are: (1) manual, (2) automatic and (3) semi-automatic. The choice of the category impacts both the segmentation process and the immediate quality of the outcomes. Furthermore, it is closely aligned with the specific requirements of the task at hand, where the degree of human involvement should be adjusted according to the precision and scale needed. For example, in fields like medical imaging, where accuracy is paramount and human expert control is obligatory, a higher degree of user involvement is often necessary (Mosqueira-Rey et al., 2023). Conversely, in applications such as real image generation, where scalability is crucial, methods with less or no direct user involvement may be more suitable. We discuss each category in turn below.

1.3.1 Manual

Manual image segmentation requires an annotator to visually assess and delineate areas of interest in images. Typically, the annotator draws boundaries around and colour-codes specific regions directly within the images. This method relies heavily on the annotator's ability to interpret complex visual information and make judicious decisions about where boundaries between segments lie based on their understanding of the context and the nature of the task. Therefore, in high-risk domains the annotator is usually a domain expert.

Advantages: This method generally offers highest accuracy and detail.
Disadvantages: It is highly time-consuming and subject to human error, fatigue, and inconsistency in annotation quality across different individuals or sessions.

1.3.2 Automatic

Automatic image segmentation predominantly employs ML algorithms, and especially DL models such as Convolutional Neural Network (CNN), to analyse and segment images autonomously, without the need for human intervention (Yu et al., 2023). These models are trained on large datasets to learn where to draw segment boundaries, completely automating the segmentation process.

Advantages: This method is fast and scalable, capable of processing large batches of images far more quickly than human annotators.
Disadvantages: Even the most robust algorithms may produce errors. However, the automatic methods lack the possibility to correct these errors, which is not acceptable in high-risk domains, such as medical imaging.

1.3.3 Interactive (semi-automatic)

Interactive or semi-automatic image segmentation typically involves an automatic algorithm under user control. In a simple setup, the algorithm may suggest a segmentation, which the expert may then review and correct, if necessary. This, however, is suboptimal, since the correction process may not benefit from the same algorithm, forcing the expert to refine the initial

segmentation proposal manually (Mosqueira-Rey et al., 2023). Therefore, in more advanced systems, the way the user provides the input is often changed. The time-consuming boundary drawing may be replaced with simpler scribbles or clicks, which are used to indicate the region of interest. With these inputs as a direct guidance, an advanced interactive algorithm then produces a complete annotation. This allows to combine human expertise with automated assistance, in a way, which simplifies segmentation process, but retains human control at the same time.

Advantages: This approach balances efficiency and precision, harnessing the speed of automated methods while incorporating crucial human control to ensure accuracy.
Disadvantages: The need for human intervention can still be time-consuming. Thus, finding the optimal balance between automation and interactivity is crucial.

Choosing the appropriate segmentation method depends on balancing the requirements for precision, speed, and scalability of the application. Interactive segmentation methods, which benefit from both expert control and automation provided by ML algorithms, are particularly effective in a clinical environment, where accuracy cannot be compromised.

1.4 Image Segmentation by Approach Type

Image segmentation has experienced significant evolution over the years. We can broadly categorise image segmentation algorithms into three groups based on the nature of the underlying algorithms: classical, neural, and hybrid (Yu et al., 2023).

1.4.1 Classical

Classical image segmentation methods are rooted in techniques that explicitly analyse the intensity, colour, texture, and continuity of the image data. They use handcrafted features to produce segmentations. Common methods include thresholding (Otsu et al., 1975), region growing (Nock and Nielsen, 2004), edge detection techniques (Canny, 1986), k-means clustering (Dhanachandra et al., 2015), watershed algorithms (Najman and Schmitt, 1994), graph cuts (Boykov and Jolly, 2001), random walker (Grady, 2006a), Geodesic Image Segmentation (GEOS) (Criminisi et al., 2008), random forest (Lindner et al., 2013), Suzuki-Abe algorithms (Suzuki et al., 1985) and semantic texton forests (Johnson and Shotton, 2010). For reference, segmentation results of 5 classical methods are shown in figure 1.8. Due to the handcrafted features, these methods are often interactive by design, allowing users to set parameters such as intensity thresholds or connectivity criteria, which allows them to guide the segmentation process directly.

Classical methods deliver satisfactory performance in simple scenarios. They are usually made available along with manual segmentation instruments in image segmentation software (Yu et al., 2023). However, as image acquisition technology continues to advance, the complexity of image details and overall variability of image data have significantly increased. The need to perform segmentation in specialised domains beyond real images, such as thermal imaging, hyperspectral scans, and seismic data, presents significant challenges. Advanced segmentation tasks in these modalities frequently encounter complex structures with poorly defined contours and overlapping elements, further complicated by noise and artefacts. These complexities demand segmen-

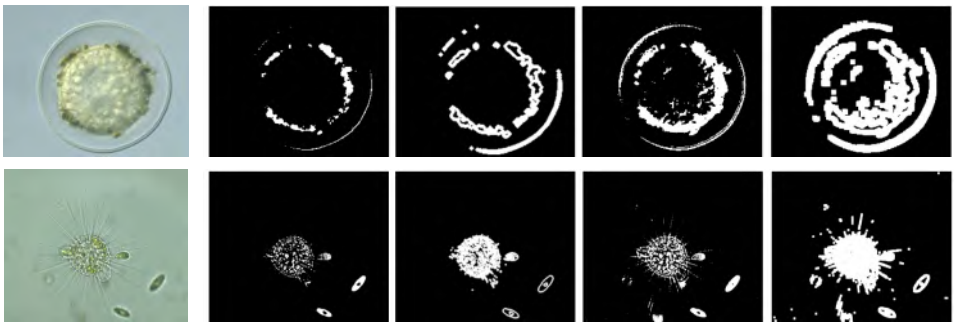


Figure 1.8: Segmentation results for 4 classical segmentation methods on microscopic cell images. Image source: (Hussain, 2024)

tation models that are not only robust but also highly adaptive, capable of distinguishing subtle nuances despite these complicating factors. Traditional feature extraction methods based on handcrafted rules fall short of addressing these demands, making them either completely ineffective or prolonging segmentation time.

1.4.2 Neural

Neural methods for image segmentation predominantly utilise DL models, with several types of network architectures being especially influential. Originally, Fully Connected Neural Network (FC) was widely used for segmentation, due to its ability to learn features across the entire input space. However, its limitation of requiring fixed-size, small images led to the development of the patch classification algorithm for image segmentation (Li et al., 2014). Further, full connections were replaced with convolutions, allowing the input of images of any size and improving performance. This resulted in a CNN (LeCun et al., 1989) being applied to various tasks, and its subtype, Fully Convolutional Neural Network (FCN) (Long et al., 2015), mainly being used for segmentation. Now, CNNs remain the most common network architecture, designed to process pixel data and effectively capture image features. However, many architectures were proposed, which approach learning from different angles. The notable among them are: FCN-based encoder-decoder architecture (Ronneberger et al., 2015), Recurrent Neural Network (RNN) (Sherstinsky, 2020) and Transformer (Vaswani, 2017), based on the concept of attention (Bahdanau et al., 2014). A number of network architecture concepts were proposed, such as multi-scale feature extraction (He et al., 2015), skip connections (Wiener, 2019) and dilated convolution (Chen et al., 2014). The field continues to rapidly evolve, with one of the latest advances being SAM 2 (Ravi et al., 2024), showing a significant improvement over SOTA.

Developing interactive systems that integrate DL poses significant challenges. Unlike classical methods with controllable parameters, DL models are not initially designed for interactive use. To enable a typical user workflow, which allows to iteratively correct the result until satisfaction, DL methods necessitate a feedback loop, where the outputs of the model are continually reintroduced as inputs. This is not trivial, since DL methods are usually trained on static datasets and may struggle to adapt to dynamic user interactions in real-time. They are thus more difficult to adapt for applications requiring user feedback.

1.4.3 Hybrid

Hybrid approaches integrate neural networks with classical methods. The resulting methods usually aim to harness DL for the segmentation, while classical algorithms are used for refinement (Chen and Pan, 2018) or regularisation.

Refinement. Basic hybrid systems might use a neural network to propose a segmentation and a classical method like graph cuts to refine these proposals with or without user feedback (Chen and Pan, 2018; Lu et al., 2018). The success of such methods strongly depends on how effectively the classical components are integrated to complement the neural network outputs. Furthermore, it is very probable that segmentations produced by SOTA neural approaches might degrade as a result of such refinement as shown in (Aflalo et al., 2022). To combat this, more advanced hybrid systems adopt end-to-end approaches, such as in (Xie et al., 2023).

Regularisation. Regularisation in ML refers to a set of techniques used to prevent overfitting by adding additional information or constraints to a model. The goal of regularisation is to improve the model's ability to generalise to new, unseen data. This is done by penalising overly complex models or by incorporating prior knowledge, often provided by classical methods, thereby promoting simpler and more robust solutions. A common approach involves incorporating a regularisation term into the loss function (Nosrati and Hamarneh, 2016). In the context of image segmentation, various forms of prior knowledge can be integrated into DL frameworks, including adjacency rules (Ganaye et al., 2019), shape constraints (Oktay et al., 2018) or topology specifications (Keshwani et al., 2020).

Currently, as it can be seen in both recent (Bilic et al., 2023) and older (Antonelli et al., 2022) challenges, featuring a large number of competing methods, hybrid methods are rarely encountered both among the competing and the top-scoring methods. This is further evidenced by a very recent SAM 2 SOTA comparison (Ravi et al., 2024), where all of the competitors for video object segmentation are purely DL-based. This may indicate that hybrid methods systematically underperform or introduce additional computational complexity, as with graph cut, which is unfeasible. However, hybrid methods find more success in complex domains, where including prior knowledge is essential for many problems, such as in medical imaging segmentation (Lepcha et al., 2023).

1.5 Medical Imaging

Medical imaging techniques utilise various physical principles like radioactivity, electromagnetic waves, nuclear magnetic resonance, and sound waves to non-invasively create visual representations of the internal structures of the human body (Suetens, 2017). The primary imaging techniques used in clinical settings are X-ray radiography, Computed Tomography (CT), MRI, PET, and Ultrasound (US). Overall, medical imaging contributes to approximately 90% of all healthcare data (Zhou et al., 2021), making it one of the most important sources for medical data analysis.

1.5.1 Types

We give an overview of each of the primary medical imaging techniques below. Examples of the respective medical scans are shown in figure 1.9.

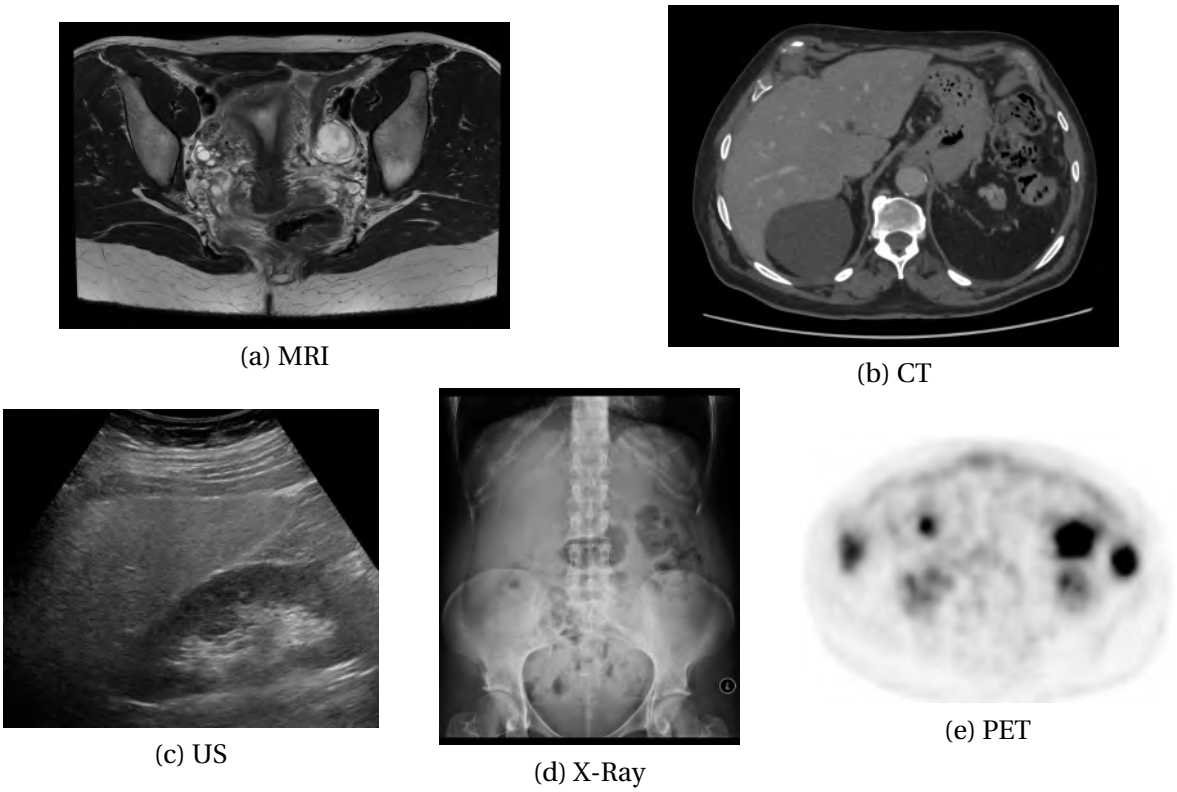
X-ray (Radiography). Radiography uses X-rays, a form of electromagnetic radiation, to capture images of the internal structures of the body by directing a beam of energy at the specific body part being examined (Suetens, 2017). When X-rays pass through the body, they are absorbed at different rates by different tissues. This allows for the visualisation of bones, certain tumours, and lung conditions using an X-ray detector placed on the opposite side of the body part being examined. Radiography is widely used for diagnosing fractures, infections, and detecting foreign objects in the body.

Computed Tomography (CT). CT scan utilises the X-ray technology to create cross-sectional images of the body. As opposed to radiography, it generates a 3D image of internal organs and structures, by combining multiple x-ray measurements taken from different angles. More precisely, a CT scan involves rotating an X-ray beam around the patient while moving along the patient's body, producing a set of cross-sectional images, also known as slices or tomographic images. Once the desired number of slices is acquired, they are digitally stacked together into a 3D image of the body part in question. A CT scan can provide detailed images of various structures, such as bones, muscles, organs, and blood vessels. CT scans are useful for detecting various injuries and diseases, including certain types of cancers, and cardiovascular diseases.

Magnetic Resonance Imaging (MRI). In contrast to CT, MRI uses strong magnetic fields and radio waves to produce detailed 3D images of organs and tissues. More precisely, an MRI machine uses strong magnets to create a powerful magnetic field that aligns the protons in the body with this field. When a radiofrequency pulse is applied, these protons become excited and move out of their aligned state. Then, once the radiofrequency pulse is turned off, the protons return to their original alignment, releasing energy in the process. MRI sensors detect this released energy. The time it takes for the protons to realign and the amount of energy released vary depending on the environment and the chemical properties of the molecules. These variations help physicians differentiate between different tissue types. MRI scanners excel in imaging soft tissues in the body and do not rely on the ionising radiation used in CT scans and radiography. MRI is capable of providing clearer images of the brain, spinal cord, nerves, muscles, ligaments, tendons and organs in general.

Positron Emission Tomography (PET). PET imaging involves injecting a small amount of radioactive tracer into the patient's bloodstream. This tracer emits positrons, which interact with electrons in the body to produce gamma rays. The scanner detects these gamma rays to construct images that show metabolic activity within tissues. PET scans are often used in oncology to detect cancer and monitor its progression and treatment. A PET scan can often identify abnormal tracer metabolism associated with diseases before they appear on other imaging tests like CT or MRI.

Ultrasound (US). US imaging uses high-frequency sound waves to produce images of the patient's internal organs and structures. During a US scan, a transducer, which is a handheld device that both emits and receives sound waves, is placed on the skin. It emits sound waves into the body and detects the echoes that bounce back. These sound waves reflect off boundaries between different types of tissues, such as between fluid and soft tissue or soft tissue and bone. The



(a) MRI (b) CT (c) US (d) X-Ray (e) PET

Figure 1.9: Examples of medical scans of the abdomen: (a) MRI, (b) CT, (c) US, (d) X-Ray, and (e) PET. Images only for illustrative purposes, each scan corresponds to a different patient. Image sources: (a) FPMRId, (b) SURGAR (SURGAR, 2024) clinical trial data and (c, d, e) publicly available individual slices of medical scans (Parasher and Mohan, 2021; Jones, 2024; Holdsworth et al., 2007).

transducer picks up these echoes and converts them into electrical signals, which are sent to the US scanner. By measuring the time it takes for the echoes to return and knowing the speed of sound in tissues, the scanner calculates the distance to the tissue boundaries. These distances are used to create real-time images of tissues and organs on a monitor. US images can be displayed in 2D, 3D, or 4D (i.e. 3D images in motion). US is widely used to assess soft tissues and organs, including monitoring fetal development in obstetrics and examining the heart and blood vessels in cardiology.

1.5.2 Clinical Usage

Medical imaging is essential in clinical practice, playing a pivotal role not only in identifying diseases but also in guiding treatment and managing ongoing health conditions (Zhou et al., 2021). The basic radiology workflow features the following steps:

1. **Image Acquisition:** A technologist operates imaging equipment, such as CT or MRI machines, to capture detailed medical images of the patient. These images are converted into the Digital Imaging and Communications in Medicine (DICOM) format. This standardised format ensures that the images can be transmitted and interpreted across different systems.

2. **Image Routing:** The images are sent to a DICOM router, a system that directs the images to the appropriate destinations.

3. **Image Storage and Management:** The router forwards the images to the Picture Archiving and Communication System (PACS). PACS stores, retrieves, and manages the images, making them accessible for further use.

4. **Radiologist Access:** Physicians, especially radiologists, are primarily responsible for interpreting medical images. They use specialised workstations to access the image data stored in PACS. They visualise, post-process, and interpret the images to assist in diagnosis and treatment planning.

Radiologists typically analyse the images manually or with the help of classical tools - by marking, delineating the regions of interest and leaving notes. They then prepare a detailed report summarising their observations. The referring physician then uses this report, along with the images, to develop a diagnosis and create a treatment plan. The role of radiology in healthcare can be categorised into several key directions: prevention, detection, diagnosis, delivery and monitoring of therapy, prognosis, and other considerations (Brady et al., 2021):

Prevention. Radiology significantly contributes to disease prevention through screening and predictive imaging biomarkers, facilitating early detection. It provides reassurance by confirming the absence of disease, which can eliminate the need for further costly tests.

Detection. Early disease detection is enhanced through population-based screening programs in radiology, allowing for the identification of diseases across large populations. By spotting abnormalities that match clinical symptoms, radiology ensures patients receive timely diagnosis.

Diagnosis. Imaging is crucial for determining disease stages, which is essential for planning effective treatment strategies. Accurate staging and decision-making are heavily dependent on accurate expert interpretation.

Delivery and Monitoring of Therapy. Tracking patient progress during treatment is vital for assessing treatment effectiveness, distinguishing between those who respond well to treatment and those who do not.

Prognosis. Confirming disease resolution is crucial for determining when treatment can be safely discontinued.

Medical imaging is frequently used during patient follow-ups to assess the effectiveness of treatments. Furthermore, medical images are increasingly vital in invasive procedures, being used directly for surgical planning. Therefore, accurate interpretation of medical images is important for effective patient care, as it directly influences diagnosis, treatment planning, and outcomes. It is evident that skilled radiologists are essential for providing precise interpretations, which involve analysing complex imaging data to identify abnormalities, determine disease stages, and evaluate treatment responses. However, this process is time-consuming and is subject to human limitations. Specifically, human interpretation is often limited by subjectivity, variability among interpreters, and fatigue. Radiologists normally have limited time to review a rapidly increasing volume of images, potentially resulting in errors and imprecisions, which is not feasible.

1.5.3 Specifics & Challenges

The challenges and time-intensive nature of interpreting, annotating, and incorporating medical images into algorithms arise primarily from the following factors (Zhou et al., 2021):

Quantity of modalities. Medical images come in various modalities, which are very different from one another. New imaging techniques, such as spectral CT, are continuously being developed. Each modality requires experts in a specific field.

Image heterogeneity. Images are diverse due to differences in equipment, scanning protocols, and patient characteristics, causing data distribution shifts. Simply, even data which appears similar to the eye, might produce very different results if input in the same algorithm. Patient privacy and data management practices lead to images being stored in disparate locations, making centralised data sets rare.

Diversity of pathologies. Radiology Gamuts Ontology (Budovec et al., 2014) lists 1674 differential diagnoses, 19.017 terms, and 52.976 links between terms. Provided such a quantity, disease incidence shows a long-tailed distribution, where common diseases have ample data, but rare diseases lack sufficient cases for analysis.

Data noise and sparsity. Medical images may be annotated differently, depending on the task. For example, when the task is to confirm if a tumour is present or not, a classification label in text can be provided. However, if the task is to segment the said tumour, one has to carefully draw the tumour's boundary, which is more labour-intensive. These annotation types are not immediately convertible one into another, and multiple of them may be required at the same time. This is costly and time-consuming, leading to sparse (i.e. incomplete or missing) annotations. Furthermore, because of the human factor, different levels of experience between experts and variable conditions, both inter- and intra-expert annotation variability is high (Lecart, 2024). Establishing a gold standard is thus non-trivial.

Data imbalance. Medical images show significant appearance variation, creating a multi-modal distribution of positive and negative samples. There is a pronounced imbalance between positive and negative samples, e.g., tumour pixels may take a fraction of the overall 3D volume.

Tasks diversity. Medical imaging involves a wide range of complex tasks including reconstruction, enhancement, restoration, classification, detection, segmentation, and registration. Combining these tasks, multiple imaging modalities and diverse pathology types results in numerous complex applications with unique challenges to be addressed.

Ethical and legal considerations. As opposed to other types of data, medical images contain sensitive information. Therefore, regulatory and compliance requirements have to be respected to obtain and use medical images. They include: ensuring patient privacy through data anonymization (Olatunji et al., 2021), obtaining informed consent on data usage and addressing bias to ensure fairness across diverse populations (Albahri et al., 2023).

The complexities inherent in medical imaging, including the diverse modalities, image heterogeneity, the wide range of pathologies, and legal considerations present significant challenges in efficiently interpreting and utilising this data for clinical purposes. These challenges highlight the need for solutions that can assist medical experts in medical image analysis. This need might be answered by ML and, specifically, DL.

1.6 ML for Medical Imaging

Although Computer-aided Detection (CADe) systems, Computer-aided Diagnosis (CADx) systems and decision support systems in general have been utilised in medical imaging for the last several decades, significant progress in clinical decision-making has been achieved more recently due to the advent of ML and notably DL in many tasks (Najjar, 2023). This progress is attributed to advancements in computational power and improved electronic access to clinical data. Still, no single algorithm can universally apply to all medical imaging tasks. Therefore, each method must be carefully tailored to the specific task modality, conditions and challenges to ensure accurate and reliable results (Zhou et al., 2023b).

1.6.1 General applications

ML in medical imaging can be broadly categorised into four key areas, each addressing different aspects of healthcare improvement: (1) medical image analysis and interpretation, (2) operational efficiency, (3) predictive and personalised healthcare, and (4) clinical decision support (Khalifa and Albadawy, 2024). The first area focuses on enhancing image analysis, where it helps in identifying objects of interest and reducing human error. The second area targets improving operational efficiency by speeding up the process of image interpretation and making it more cost-effective. The third area involves predictive and personalised healthcare, utilising historical data for early diagnosis and tailoring treatments based on individual patient characteristics. Finally, ML supports clinical decision-making, especially in complex procedures, including surgical interventions, and integrates with other technologies, like electronic health records.

Medical imaging has given rise to several important research domains, including:

Medical image reconstruction (Ahishakiye et al., 2021) involves converting signals captured by devices like CT or MRI scanners into images. This process is crucial for generating high-quality images even with lower doses of radiation or quicker scans. For example, GE HealthCare has produced and integrated AIR Recon DL (Electric, 2022) for MRI. This DL-driven reconstruction method has improved image clarity via denoising and sharpening by up to 60% and reduced scan times up to 50%, and is used across a broad range of MRI systems, including older models.

Medical image enhancement (Lepcha et al., 2023) targets improving image quality, which includes super-resolution, denoising and MR bias field correction (Tustison et al., 2010b). Recent advances feature image generation and modality-to-modality translation (e.g. CT to MRI), which can potentially mitigate the limited data regime typical for medical imaging (Özbey et al., 2023;

Kazerouni et al., 2023).

Medical image object detection (Jaeger et al., 2020; Chan et al., 2020) focuses on localization (CADe) and identification (CADx) of specific objects or regions of interest within medical images, when pixel precision is not required. An challenging example of this is detection of endometriosis in laparoscopic surgery videos (Leibetseder et al., 2022). Notably, object detection may serve the purpose of finding the anatomical landmarks, which will be used as inputs to another algorithm.

Medical image segmentation (Qureshi et al., 2023) is the process of extracting the boundaries of pathological or anatomical structures within medical images. This technique offers more precise object localization than detection methods, making it indispensable in scenarios where high accuracy is critical. Segmentation plays a crucial role in various aspects of medical image analysis, including tumour localization, tissue and organ quantification, disease progression monitoring, and, importantly, surgical planning (Conze et al., 2023).

Medical image registration (Hering et al., 2022) is the process of aligning multiple images into a common coordinate framework, which is essential for comparison and analysis across different multiple modalities. For instance, aligning a patient's CT scan with their MRI allows clinicians to combine detailed anatomical data from the CT with functional or metabolic information from the MRI. It is widely used in areas like multimodal fusion, and in segmentation via label transfer approaches.

Other technologies include image or view recognition (Xu et al., 2018; Chauhan et al., 2022), which targets classifying image perspectives or views; and automatic report generation (Zhou, 2023), which streamlines the process of creating diagnostic reports.

1.6.2 Specifics & Challenges

Application of ML in medical imaging is limited by the three key factors:

Limited data availability. Medical imaging datasets are often limited due to patient privacy concerns, the high cost of acquiring annotated data, and the rarity of certain conditions (Tajbakhsh et al., 2020). Specifically, due to the resource-intensiveness of data annotation, the quantity of annotated data is normally just a fraction of all the non-annotated data due to the years of patient records not yet used for any algorithm. This scarcity of annotated data makes it challenging to train robust ML models, which typically require large, diverse datasets to achieve high performance and generalizability. While there is no immediate solution to this problem, many methods have been proposed to address limited data in medical imaging (Adadi, 2021). This includes methods to create additional data such as data augmentation and generation (Chen et al., 2021b; Murtaza et al., 2023), as well as learning from limited samples using techniques like transfer learning (Iman et al., 2023) and few-shot learning (Song et al., 2023).

Critical nature of clinical decisions. While ML models often achieve superhuman performance in certain tasks, they are still error-prone. However, in medical imaging an incorrect diagnosis or

treatment recommendation can have severe consequences, including patient harm. Therefore, it's crucial that ML models not only achieve high levels of accuracy and reliability but also be designed to work in tandem with healthcare professionals, allowing for user control. This means that to integrate ML models into the decision-making process, the most adapted approach is Human-in-the-loop (HITL) (Wu et al., 2021; Budd et al., 2019). Simply, medical experts should be able to correct ML outputs, if required.

Explainability. ML models, particularly DL models, generally function as 'black boxes', which input data and output a decision, whereas the connection between one and the other is not always clear, making their decision-making process difficult to interpret (Van der Velden et al., 2022). However, it is crucial that medical experts understand the reasoning behind ML model decisions, especially in complex cases. Simply, ML models should operate in a predictable, explainable manner. The lack of explainability in these models presents a significant barrier to their adoption in clinical settings, where transparent and interpretable decisions are crucial for safe patient care and regulatory approval.

1.7 Medical Image Segmentation

Segmentation is one of the key tasks for medical image interpretation. However, delineating precise boundaries within medical images is a difficult, time-consuming task for medical experts. Furthermore, medical imaging is a challenging domain with a number of limitations, making it a research challenge for potential ML approaches. Nevertheless, producing clinically-adapted segmentation solutions is crucial for reducing the radiologists' workload. Automating this process will lead to new and more efficient medical imaging workflows, allowing medical experts to focus on other aspects of patient care.

1.7.1 Difference from Other Segmentation Tasks

Medical image segmentation is challenging due to a combination of two groups of factors, which are: (1) data-related factors and (2) clinical practice- and environment-related factors. We discuss each of these in turn.

Data-related factors. Medical image segmentation inherits the 3 major challenges stemming from medical imaging data at large, which are discussed in section 1.5.3. First, unlike typical 2D segmentation, medical imaging often deals with 3D data, which introduces complexities in handling volumetric information. Second, the variability in image quality, anatomical complexity, and the high degree of inter-expert variability—stemming from differences in expertise and annotation conditions—further complicate the process. Third, privacy and security concerns limit data availability, making centralised datasets rare with particularly small amounts of training data for rare pathologies.

Clinical practice- and environment-related factors. Medical image segmentation approaches are conditioned by the specifics of the clinical practice and clinical environment. Specifically, there are 2 key aspects, which have to be considered for any medical image segmentation solution and

are discussed in section 1.6.2. First, fully-automatic segmentation algorithms generally can not be employed in high-risk applications, such as medical imaging, due to potential errors. Second, segmentation algorithms should operate in a predictable, repeatable and explainable manner. Hence, interactive approaches are preferable in order to prevent patient harm and due to regulatory reasons. Simply, medical experts should be able to correct the outputs of such a segmentation algorithm when necessary.

1.7.2 Solutions

Medical image segmentation inherits its categorisation from general Image segmentation discussed in section 1.4. Thus, medical image segmentation approaches can be roughly divided into 3 categories, depending on the nature of the algorithm: (1) classical, (2) neural and (3) hybrid.

Classical. Prior to the rise of learning techniques, medical image segmentation relied heavily on mathematical models and low-level image processing, particularly statistical shape models, which were often enhanced using prior shape knowledge (Sharma and Aggarwal, 2010; Conze et al., 2023). These methods include region-based, classification and clustering techniques, as well as atlas-based and model-based techniques. In atlas-based segmentation, comprehensive anatomical, size, shape, and feature data of one or multiple objects of interest are combined into an atlas. Registration is then employed to adapt and apply the best-fitting reference from the atlas, to a patient scan. In contrast, model-based segmentation utilises the consistent geometric patterns of organs, applying probabilistic models to accommodate their variations in shape and structure. Some well-known examples of classical methods are Otsu's thresholding (Otsu, 1979), k-nearest neighbour (Fix, 1985), fuzzy C-means (Bezdek et al., 1984), graph cut (Boykov and Jolly, 2001) and multi-atlas segmentation (Aljabar et al., 2009).

However, these classical methods face significant challenges when dealing with the inherent complexities of medical images (Conze et al., 2023). The variability in anatomical shapes between patients, indistinct boundaries, and varying tissue characteristics present significant challenges. Moreover, the robustness of these methods is frequently compromised by common issues in medical imaging, such as noise, inconsistent contrast levels, and a variety of artifacts. While classical methods provide a solid foundation, their limitations in addressing these challenges led to advancement of ML and, specifically, DL methods.

Neural. Neural approaches use ML and, specifically, DL to perform segmentation. The most widely-used models in this domain are based on CNNs. To adapt to the volumetric nature of medical data, 3D CNNs(Tran et al., 2015) have been developed, enabling the direct processing of 3D medical images. 3D CNNs improve performance when applied to volumetric data like CT or MRI scans. The U-Net architecture (Ronneberger et al., 2015) has become particularly prominent due to its ability to produce detailed segmentations, which is crucial in medical applications. U-Net has given rise to numerous variations, including its 3D counterpart, 3D U-Net (Çiçek et al., 2016). However, given the variability of medical imaging tasks and the difficulty of transferring a model optimal for one task to another, there is a growing need for adaptable frameworks that can automatically optimise for different datasets. Prominent example of such a model are nnU-Net (Isensee et al., 2021) and NiftyNet (Gibson et al., 2018). While U-Net and its variants

remain central due to their versatility and performance, there is a growing interest in exploring other architectures, such as Transformer-based models (Vaswani, 2017), which is an emerging trend (Conze et al., 2023).

Hybrid. While classical segmentation methods have been foundational in image analysis, their application in medical imaging segmentation is often limited to simpler cases or requires extensive manual correction. However, as with general image segmentation, they can be integrated into DL frameworks to refine the ML outputs (Luo et al., 2021) or for regularisation of ML learning, which includes providing explicit constraints (Bonfiglio et al., 2023) and embedding prior knowledge (Xie et al., 2021). This latter is especially effective, since medical knowledge is the cornerstone of medical imaging and is generally obtained on a very large number of cases, which are validated by numerous medical experts over the years of practice. Usage of such data in various forms has been shown to improve segmentation performance for medical imaging. For instance, shape priors are frequently used to ensure that segmentation outputs are anatomically plausible (Boutillon et al., 2022). Beyond shape, other geometric attributes such as size, texture or topology, can be incorporated into DL training objectives (Conze et al., 2023).

1.8 Applications

Applications of image segmentation vary significantly depending on the specific domain and the particular problem being addressed, each with its own set of challenges and requirements. For instance, certain examples and some of the respective challenges are: (1) autonomous driving - due to numerous input modalities, varying conditions and lack of accuracy guarantees (Chen et al., 2024), (2) agriculture for crop and soil monitoring - due to similarity of disease stages and complexity of the agricultural environment (Lei et al., 2024), (3) industrial inspection - due to noise, dust and vibrations, as well as fast computation time requirements and the integration difficulties (Usamentiaga et al., 2022). This applies to all domains and, specifically, to medical imaging.

We introduce the applications relevant to this work and their specifics in the corresponding sections below. Namely, they are: (1) enabling AR in gynaecological laparoscopy and (2) data-efficient annotation at scale applied to segmentation.

1.8.1 Specific Segmentation Applications

Augmented Reality for Laparoscopic Surgery

Laparoscopic surgery is a minimally invasive procedure performed through small incisions using a laparoscope with a camera and specialised thin instruments. It has significantly advanced surgical techniques by reducing recovery times and minimising patient trauma. A key research focus within Computer-Aided Intervention (CAI) is the development of AR systems to assist laparoscopic surgery teams by providing enhanced visual guidance. Specifically, over the past two decades, AR has garnered significant interest in Minimally Invasive Surgery (MIS) due to the possibility to overlay diverse information, such as preoperative 3D models from CT or MRI data, resection paths, and Laparoscopic Ultrasound (LUS) images, directly onto intraoperative visuals. Integration of AR simplifies surgical procedures by eliminating the need for surgeons to mentally associate separate data sources, such as an MRI scan, with the intraoperative scene

Figure 1.10: An example of operating room setup during laparoscopic surgery. Image source: (Minig, 2024)

while operating. While this enhances precision and efficiency across various MIS specialities, additional preoperative steps are typically necessary. Specifically, the CT or MRI data must first be segmented to produce the 3D models, which are then overlaid onto the intraoperative video feed.

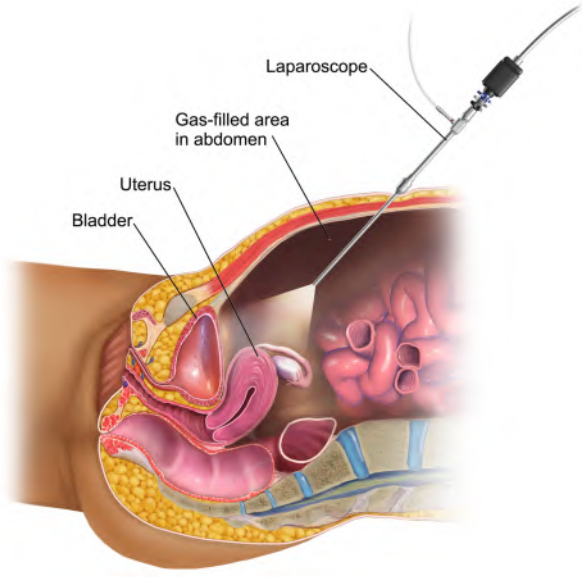


Figure 1.11: Patient-side schematic representation of the laparoscopic surgery of the uterus. Image source: (Blausen.com staff, 2014)

Minimally Invasive Surgery (MIS). As opposed to open surgery, which involves a large incision in the abdominal wall, MIS is a family of surgical techniques, which involves performing operations through small incisions or natural body orifices. MIS encompasses a variety of techniques tailored to different organs and regions of the body. These procedures utilise specialised thin instruments, including an endoscope, which is equipped with a camera and light source to visually examine the interior of a body cavity or organ. Typically, the surgeon navigates the surgical area using the endoscope inserted through one of the incisions, with the camera feed displayed on an external monitor (Robinson and Stiegmann, 2004, 2007). Because of this indirect approach, extensive and

specialised training, which can take years to complete, is required to achieve proficiency.

Laparoscopy, also known as keyhole surgery, is one of the first forms of MIS and targets the abdomen and pelvis. The procedure involves inflating the abdominal cavity with gas and inserting a rigid telescope (laparoscope) through a small incision in the abdominal wall to view the peritoneal cavity. Once the laparoscope is in place, additional surgical instruments are introduced through nearby incisions to perform diagnostic or surgical tasks (Monnet and Twedt, 2003). Laparoscopy generally goes through these steps (Smith et al., 2018):

1. A small incision is made in the peritoneal cavity, and a trocar (hollow tube) is inserted providing access inside for the instruments.
2. Carbon dioxide is introduced to inflate the abdominal cavity to enlarge the working space and obtain better visibility.
3. A laparoscope is inserted, and the scene is explored.
4. Trocars for other instruments are placed as needed.
5. The surgery is performed by driving instruments through the trocars.
6. Instruments and trocars are removed, the carbon dioxide is released, and incisions are closed.

Figure 1.10 depicts the operating room setup during laparoscopic surgery, where surgeons monitor internal views on screens using a laparoscopic camera. The schematic representation of the procedure within the gas-filled abdomen of the patient is shown in figure 1.11.

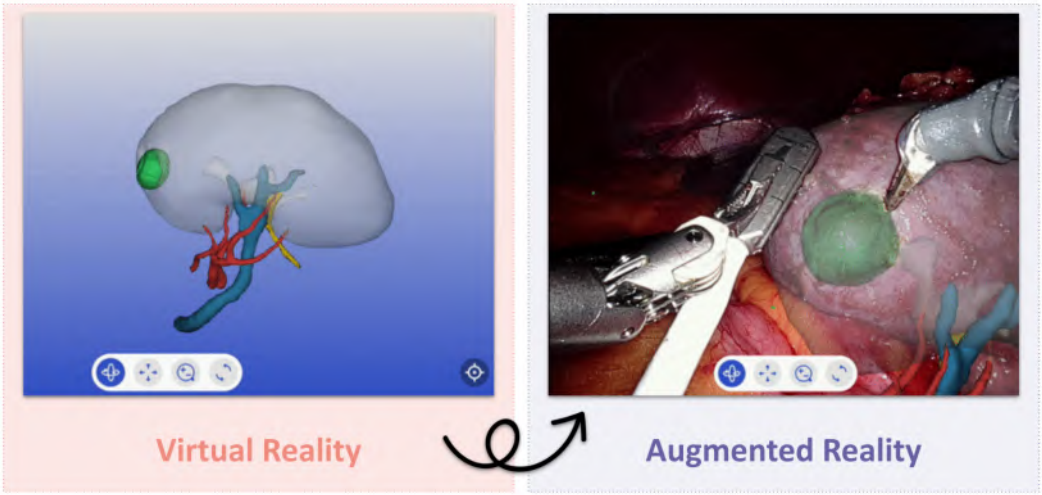


Figure 1.12: Comparison between Virtual Reality (VR) and AR: (left) a 3D digital model of a kidney; (right) the same 3D model overlaid onto a surgical video frame. The left image displays only digital information, while the right image combines digital content with real-world visual data.

Augmented Reality. AR is a set of technologies, which targets seamlessly blending digital information with the physical world, either automatically or through user interaction (Arena et al., 2022). Specifically, AR allows to combine and overlay digital content and information (e.g. text, images, audio, video, graphics or any other digital format) onto the real-world environment

in real time. AR allows users to interact with a digitally enriched version of their surroundings, using various display technologies that combine or superimpose this digital content onto their view of the physical world. AR may be realised in a software-only form or require additional specialised hardware, like a virtual reality headset (Angelov et al., 2020), depending on the purpose and application. Simply, VR implies only digital information, while AR combines digital and real-world information. Figure 1.12 provides a simple illustration of the differences between the two, based on the AR for the laparoscopic surgery of the kidney.

Augmented Reality for Laparoscopic Surgery of the Uterus. CAI is a multidisciplinary field that focuses on the use of computational technologies to assist in planning, guiding, and executing surgical and interventional procedures. A major research focus in CAI is enhancing laparoscopic surgery through AR guidance. This approach integrates data from modalities like MRI with real-time video coming from the laparoscope, allowing surgeons to visualise internal anatomical structures invisible to the naked eye during the procedure.

Figure 1.13: The operating room screens during the laparoscopic surgery of the uterus using an industrial system U-SURGAR (SURGAR, 2024), which stems from (Collins et al., 2020). Image source: (SURGAR, 2024)

The system proposed in (Collins et al., 2020) is the first AR-guided approach for laparoscopic surgery of the uterus, called Uteraug. It was then adopted and advanced into an industrial system U-SURGAR (SURGAR, 2024). The system operates using preoperative MR or CT data combined with monocular laparoscopes, eliminating the need for additional interventional hardware such as optical trackers. As input, it requires a segmented preoperative 3D model of the uterus, including both the surface mesh and the meshes of internal structures. These meshes are then semi-automatically aligned with the uterus in the surgeon's visual field and tracked, achieving a see-through effect for the respective anatomical structures. Simply, the tumours inside the uterus are visible on the surgeon's screen with the laparoscope video feed, as if the uterus was transparent. The proposed AR system operates on a dedicated hardware platform and is compatible with standard monocular or 3D laparoscopes, as well as surgical robots. The

main phases of the pipeline are showcased in figure 1.14. The operating room screens while using U-SURGAR are shown in figure 1.13. To produce the 3D model of the uterus and its internal structures for AR visualisation, a preoperative MRI scan of the patient's pelvis is first required. This scan then must be segmented by a radiologist, so that the model could be reconstructed from the segmentations.The work presented in this thesis addresses segmentation both as an integral part of U-SURGAR (see section 4) and as a standalone task. It also takes a step further and adopts a broader view by proposing a general data-efficient annotation method applicable to a wide range of scenarios (see section 5).

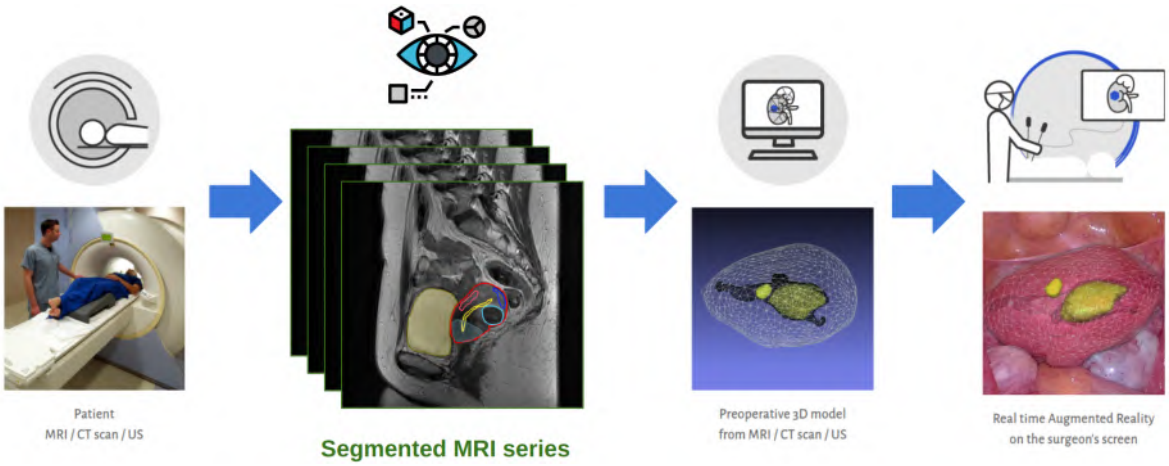
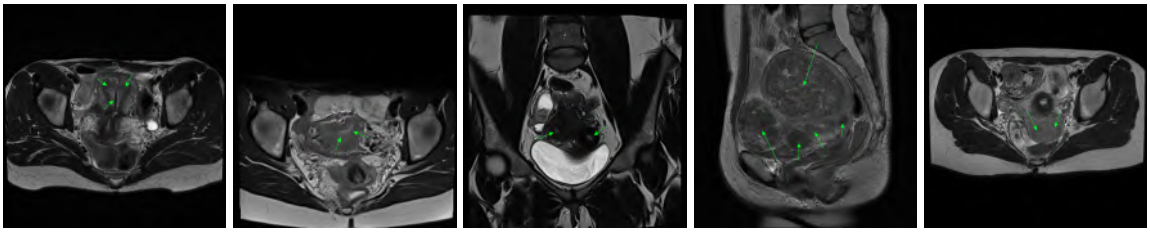


Figure 1.14: The outline of the AR pipeline for laparoscopic surgery in U-SURGAR, which builds upon the Uteraug system described in (Collins et al., 2020). As one of the system's applications, the uterus is first shown in an MRI (Step 2) and then in a surgical video (Step 4). Segmented MRI series is highlighted in green as the key area of this thesis contributions.

FPMRI. MRI is considered the gold standard imaging modality for pelvic examinations in both adults and children (Virzì et al., 2020). MRI imaging of the female pelvis is a valuable tool for distinguishing between non-cancerous (benign) and cancerous (malignant) masses, determining the stage of cancer before treatment, and assessing other gynecologic and pelvic conditions. Its ability to produce high-contrast images of soft tissues makes it especially effective for identifying the spread of tumours and examining the entire pelvic area (Westbrook and Talbot, 2018). The key imaging techniques involved in MRI of the female pelvis include T2-weighted imaging (T2WI), T1-weighted imaging (T1WI), in-phase and opposed-phase imaging, as well as advanced imaging methods such as Diffusion-weighted Imaging (DWI) and Dynamic Contrast-enhanced Imaging (DCE) (Sakala et al., 2020). Samples of FPMRI T2WI are shown in figure 1.15.

As discussed in section 1.5.3, MRI segmentation presents inherent challenges. Segmentation of the female pelvis in MRI is particularly difficult due to four specific inherent factors. First, the complex anatomy involves multiple closely situated organs, such as the uterus, ovaries, and bladder, with soft tissues that often exhibit similar signal intensities, making it difficult to distinguish between them. Second, there are notable variations in positioning and pelvic morphology, making inter-patient variability especially pronounced (Lee et al., 2024). Third, organ motion, particularly from respiration and bowel peristalsis causes shifts and distortions in the images (Maccioni et al., 2023). Fourth, MRI images may contain a number of artefacts, which come from MRI hardware and room shielding, MRI software, patient and physiologic motion, tissue heterogeneity, presence

of foreign bodies and due to the specifics of signal processing and reconstruction techniques. This includes motion artefacts due to patient movement, susceptibility artefacts due to air in the bowel, and chemical shift artefacts, which result in blurred boundaries between tissues (Westbrook and Talbot, 2018). These difficulties are further compounded by the fact that most segmentation software tools predominantly focus on CT images, with MRI applications being largely limited to brain imaging (Virzì et al., 2020).



(1) **(2)** **(3)** **(4)** **(5)**

Figure 1.15: Female pelvis MRI samples, with main difficulties indicated with green arrows, series 1 to 5: **(1)** presence of an IUD, not seen in the training set; **(2,5)** unclear contours, blurriness of the uterine cavity; **(3)** similarity of the uterine (left) and cervix cavities (right); **(4)** strong uterus deformation due to the tumours, with here five tumours.

While previous research has demonstrated that FPMRI segmentation is feasible using ML and specifically DL (Liu et al., 2021; Zhuang et al., 2024), number of proposed approaches useable in clinical environment remains limited due to the abundance of fully-automatic solutions (Kalantar et al., 2021; Omouri et al., 2024). The problem is worsened by severe scarcity and limited size of publicly available annotated abdominal and pelvic MRI datasets. Specifically, one of the largest datasets contains only 300 scans (Pan et al., 2024).

Overall, FPMRI segmentation is challenging due to the inherent complexities of the anatomy and the current lack of sufficient public datasets (Virzì et al., 2020). These challenges are further exacerbated by the limited availability of appropriate solutions, as most existing tools are tailored for other imaging modalities or not clinically-adapted in practice. However, addressing these issues could significantly streamline radiologists' workflows and pave the way for innovative applications, such as enabling AR in the laparoscopic surgery of the uterus.

Data-efficient Annotation at Scale

Data hungriness is one of the key downsides of modern ML and DL algorithms. For example, Generative Pre-trained Transformer 3 (GPT-3) - a Large Language Model (LLM) was trained on 45TB of compressed plaintext (Brown, 2020). In view of this, to train all the more effective ML models, a large-scale annotation effort, or annotation at scale, is required. Specifically, it involves systematically annotating vast quantities of data to create extensive and diverse large training sets for the ML models. However, annotation at this scale is not feasible in all the domains due to the resource-intensiveness of the task and the need for domain-specific experts. For example, in the medical field, a typical dataset contains very limited annotated data, sometimes just several scans, and large quantities of non-annotated data, if any. The goal of the data-efficient annotation at scale is thus enabling the large-scale annotation in domains, where it's currently unfeasible. More precisely, data-efficient annotation at scale should address the problem data hungriness in two major ways: (1) by effectively utilising the limited annotated data available and (2) by anno-

tating available non-annotated data with minimal human input. Given that increasing dataset size is a key factor in performance improvement (Adadi, 2021; Halevy et al., 2009), leveraging both approaches is crucial. This is especially true for medical imaging, where the majority of data remains unannotated. A prime example is surgical video segmentation from laparoscopy, which can be used for many tasks, including carcinomatosis detection. Specifically, surgeries normally last hours, resulting in long and numerous files of video footage, containing hundreds of thousands of frames, which cannot realistically be annotated within a reasonable timeframe by human effort alone (Ward et al., 2021). Another example is the Cancer Imaging Archive (TCIA) (Clark et al., 2013), which hosts a vast amount of medical imaging scans, with only fractions of its volume periodically annotated.

1.9 Contributions

The goal of this work is twofold.

First, we enable FPMRI segmentation with a new dataset and a clinically-adapted solution, which improves performance by incorporating the way a medical expert typically approaches segmentation. Simply, existing methods do not exploit the typical sequentiality of real user interactions. This is due to the interaction memory used in these systems, which discards ordering of user interactions. In contrast, we show that the order of the user corrections should be used for training of DL models and leads to performance improvements. With this, each subsequent correction done by a medical expert draws from the corrections provided so far and their corresponding results. This approach ensures continuity, enabling higher segmentation precision with fewer steps and reduced human input.

Second, we aim to address the problem of data-efficient annotation at scale on the example of segmentation, by integrating the three steps of data selection, annotation and training into a single architecture. This approach allows rapid annotation of data in the absence of already available neural annotation tools and reduces the overall amount of data needing annotation while maintaining performance. Specifically, this approach requires just a handful of annotated data to enable a single interactive DL model as described above, which is both an annotation tool and a predictor. Simply, under the respective technical and medical experts' control, the model improves by producing the annotations for itself, which addresses both the problem of the absence of a task-specific annotation tool and the absence of the task-specific annotated data.

Concretely, in this work we make a total of four contributions split into two groups. The first group forms the foundation of the main contributions by focusing on data and includes: (1) the FPMRI dataset and (2) the inter-expert variability study on this dataset. The second group contains the main contributions, which are application-focused: (3) an interactive FPMRI segmentation framework and (4) a framework for data-efficient annotation at scale applied to segmentation.

1.9.1 Data-wise

Contribution #1. We have assembled and annotated a FPMRI segmentation dataset with assistance of medical experts from the Centre Hospitalier Universitaire (CHU) de Clermont-Ferrand (university hospital). To our knowledge, this is the first dataset of its size that contains segmentations of the following anatomical structures: uterus, bladder, uterine cavity, cervix, fundus

and anterior wall; as well as the following annotations of pathologies: tumours, endometriosis and adenomyosis. The dataset reflects the complexity of FPMRI segmentation and features annotations, which are multi-class, multi-label, multi-instance, and multi-component. It consists of MRI scans manually annotated by expert radiologists, capturing significant variations in shape, size, and texture among these structures. Our contribution involves the dataset itself and the process of the dataset creation, which includes the data collection process, the annotation guidelines and respective challenges. The dataset was annotated using specialised platforms, and its development was tracked, including the evolution of the collection, annotation and the effects of the annotation platform changes. The dataset is now in a stable state and has been used in other contributions for training and evaluation, for a user study in contribution #3 as well as a source for the inter-expert variability study in contribution #2. This contribution is presented in chapter 3 of the thesis.

Contribution #2. We have performed a single-centre inter-expert variability study in collaboration with a medical expert. Specifically, we have done a retrospective analysis involving 10 female patients who underwent 1.5T pelvic MRI with 5 mm thick axial T2 Propeller sequences. The MRI scans were segmented by 6 radiologists with varying levels of experience, producing segmentation for uterus, bladder, uterine myomas, uterine cavity and cervix. To evaluate the correlation between the experts, based on this segmentation data, we have calculated: (1) the dice coefficient in an expert-to-expert manner and (2) volume of each anatomical structure per expert. To aggregate these data, we calculated mean and standard deviation of these metrics for each expert and across experts for each series. Furthermore, we have used the Simultaneous Truth and Performance Level Estimation (STAPLE) algorithm to create a reference golden standard-like segmentation from the segmentations of all the radiologists for each patient and evaluated their correlation with this golden standard. The study found excellent inter-expert correlation for uterine volume, very good correlation for fibroids and bladder, satisfactory correlation for the uterine cavity, and moderate correlation for the cervix. The results suggest that segmentation of the uterus is a reliable and reproducible process, supporting the potential development of automatic or semi-automatic segmentation tools. This study is a part of the medical thesis in (Lecart, 2024), and the related journal article is planned. This contribution is presented in chapter 3 of the thesis.

1.9.2 Applicative

Contribution #3. We propose a general multi-class deep learning-based interactive framework for image segmentation, which embeds a base network in a user interaction loop with a user feedback memory. We propose to model the memory explicitly as a sequence of consecutive system states, from which the features can be learned, generally learning from the segmentation refinement process. Training is a major difficulty owing to the network's input being dependent on the previous output. We adapt the network to this loop by introducing a virtual user in the training process, modelled by dynamically simulating the iterative user feedback. We evaluated our framework against existing methods on challenging multi-class segmentation tasks, including FPMRI and liver and pancreas CT segmentation, using both in-house and public datasets. A user evaluation with eleven medical professionals from related fields showed a significant reduction in annotation time when using our framework compared to traditional tools. We systematically

evaluated the influence of the number of clicks on the segmentation accuracy. A single interaction round our framework outperforms existing automatic systems with a comparable setup. We provide an ablation study and show that our framework outperforms existing interactive systems. This method has been published as a workshop (Mikhailov et al., 2022) and journal (Mikhailov et al., 2024) articles and is protected by a patent. This method is presented in chapter 4 of this thesis.

Contribution #4. We propose a framework called SAIM, which integrates the three steps of data selection, annotation and training into a single architecture. This is made possible by three key properties of SAIM in contrast with existing work: (1) SAIM uses a deep interactive predictor; hence the classical tools are not required and the annotation predictor can be pre-trained with limited data to produce quality annotations; (2) SAIM uses a single model shared between the three steps, hence the model is deployable and the annotation predictor improves as annotation progresses; (3) SAIM uses active learning to maximise the impact of each annotation on the predictor performance, making the model rapidly improve. We evaluate SAIM and compare it to existing systems in emulated annotation scenarios in an automated manner with fully-annotated segmentation datasets on five tasks: (1) on multi-class semantic MRI segmentation of the female pelvis, (2) on multi-class semantic liver CT segmentation, (3) on multi-class semantic pancreas CT segmentation, (4) on cardiac MRI segmentation, on which we validate SAIM against SOTA Semi-supervised learning (SSL) approach, (5) on natural image segmentation, on which we validate SAIM against the SOTA Self-training (ST) approach. We demonstrate SAIM in a real annotation scenario of kidney MRI segmentation with a human user. We estimate the time gain as compared to classical segmentation tools. SAIM outperforms both classical tools and SOTA approaches. With it, one can jumpstart efficient interactive annotation from limited annotated data and minimise the amount of data to annotate, while iteratively improving performance. Part of this work has been published as a workshop article (Mikhailov et al., 2023), while an extended version is pending to be submitted to a journal. Patent application pending. This method is presented in chapter 5 of this thesis.

Chapter 2

Background

2.1 Machine Learning

AI, ML and DL are often used as synonyms in the discussions regarding AI. However, these terms are not interchangeable. The three can be distinguished as follows: (1) AI is the broad field that aims to create systems capable of mimicking human intelligence, (2) ML is a subtype of AI, which is used to extract knowledge from data without explicit programming, traditionally done through simple methods, such as support vector machines, linear regression or decision trees, (3) DL is a further specialisation within ML, which uses more advanced methods, such as deep ANNs.

Three key distinctions can be identified between ML and DL (Sengupta et al., 2020). First, in the level of automation: traditional or "shallow" ML techniques often require human intervention to manually select features and classifiers, while DL automates this process, extracting features automatically. Second, in the data and computational requirements: DL performance scales with data and often requires significantly more computational power than ML. Third, in performance: DL often surpasses "shallow" ML techniques, particularly in complex tasks, such as image and speech recognition, natural language processing, data generation and autonomous driving. In contrast to ML, DL enables a widely applicable and practical approach for non-parametric, model-based learning. This allowed DL to become a dominant approach for many tasks (Buntine, 2020; Dong et al., 2021).

Despite their rapid growth, both ML and DL are still evolving fields, whose explosive expansion owes to the decades of prior research (Wang and Raj, 2017). As such, the number of research papers published per month was shown to grow exponentially and double every 23 months (Krenn et al., 2022), stabilising at approximately 242 thousand papers per year in 2022 (Maslej et al., 2024). At the moment, hundreds of AI-related papers are published every day, with 89 notable DL models released in 2023, according to (Maslej et al., 2024). However, significant further research and innovation are necessary to fully harness the DL potential, particularly in addressing challenges like data bias, data privacy, data efficiency, and interpretability. Given this rapid growth of the ML domain in general, and DL in particular, it is essential to highlight key advancements to establish a solid foundation for this thesis. As our contributions are directly and indirectly related to the DL field, we outline key DL advancements in the next section.

2.2 Deep Learning

DL is not a single method but a collection of solutions that can be applied to a wide range of problems. It is a naturally vast and multifaceted domain. DL is spread across a number of types, utilising both general and specialised network architectures, which enable learning through various methods. Furthermore, DL relies on at least 4 essential elements. They are: (1) datasets, which provide the data that fuels learning; (2) libraries, offering a range of tools, including the ones to build, train, and evaluate models; (3) computational resources, such as Graphics Processing Unit (GPU) clusters, which handle the high demands of processing; and (4) software tools, which support data annotation, model deployment and a wide array of task-specific functions. Together, all of the above contribute to the overall success of DL models across various applications. We first cover the main aspects of DL: types, architectures, and techniques. Afterward, we explore the supporting aspects: datasets, libraries, computational resources, and software tools.

2.2.1 Types

Four main types of DL can be identified (Sarker, 2021a). They are: (1) supervised learning, (2) unsupervised learning, (3) semi-supervised learning, and (4) reinforcement learning. These types are mainly distinguished by their approach to data, characterised by presence or absence of annotations (or labels), associated with each data point. Specifically, for each: (1) in supervised learning, algorithms learn a function that connects the inputs to outputs by leveraging the annotated training data consisting of input-output pairs; (2) unsupervised learning focuses on analysing data without annotations, allowing models to discover patterns and structures independently; (3) semi-supervised learning utilises a combination of annotated and non-annotated data, integrating techniques from both supervised and unsupervised methods; (4) reinforcement learning does not require annotations, but instead employs reward signals to guide learning in a trial-and-error manner.

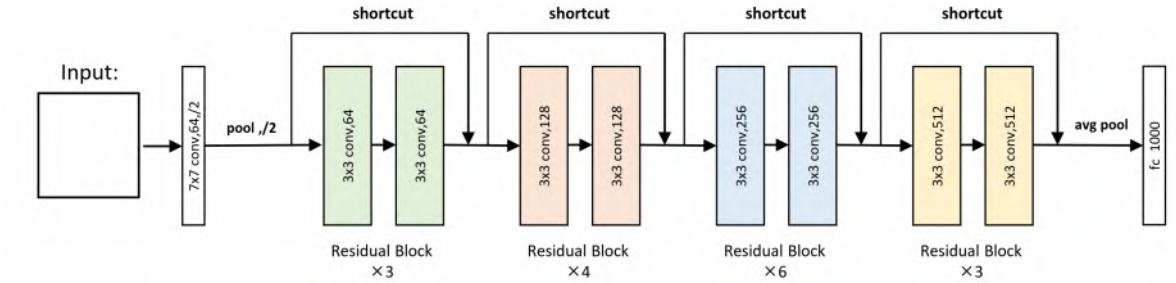


Figure 2.1: ResNet34 architecture schematic. Each box represents a set of multi-channel feature maps. Image source: (Zhang et al., 2023)

2.2.2 Architectures

In DL, architecture typically refers to the structural design of a neural network, defining how layers are organised and how data flows through the network during learning. An architecture is normally tailored to handle specific tasks or data types, enabling models to capture patterns, relationships, and representations within the data in particular ways. The choice of architecture influences a number of factors, including the model's ability to process information, the model's

generalisation to unseen data, and the model's performance. Some of the most prominent architectures are: (1) CNN, with U-Net and ResNet as examples shown in figures 2.2 and 2.1 respectively; (2) RNN, with LSTM and Gated Recurrent Unit (GRU) as examples; (3) Generative Adversarial Network (GAN); (4) Autoencoder (AE) and (5) Transformer, including the specialised Vision Transformer (ViT) (Suganyadevi et al., 2022; Dong et al., 2021; Sarker, 2021a). We cover each of these in turn.

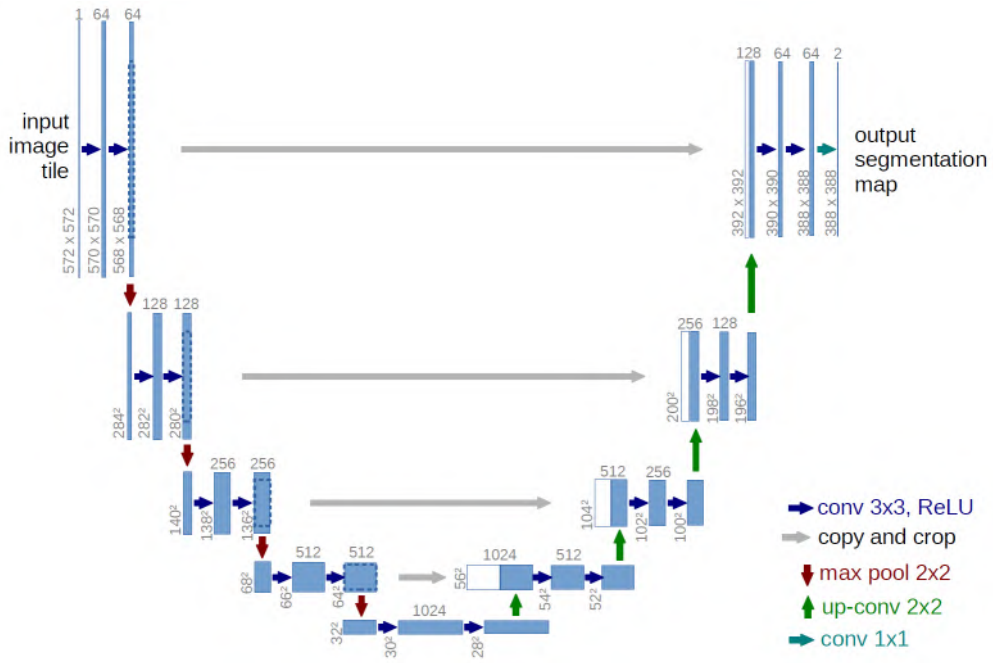


Figure 2.2: U-Net architecture schematic. Each box represents a multi-channel feature map. Image source: (Ronneberger et al., 2015)

CNN. CNNs are widely used for tasks involving gridded data, such as images. They leverage convolutional layers to automatically detect spatial hierarchies in the data, making them highly effective in computer vision tasks. CNN is a feedforward network, meaning that information flows in one direction—from the input layer to the output layer—without loops or cycles. U-Net and ResNet are two important examples of CNN. U-Net was introduced for biomedical image segmentation tasks (Ronneberger et al., 2015). Its encoder-decoder architecture, coupled with skip connections, allows the model to retain high-resolution details during upsampling, making it particularly effective for tasks requiring pixel precision, such as segmentation in medical image analysis. In turn, ResNet, or Residual Neural Network, addresses the vanishing gradient problem in DL, which is characterised by the gradual shrinking of gradient values as they are propagated back through the layers during training. This issue occurs in networks with many layers, where the gradients used to update the weights become extremely small in early layers, causing those layers to learn very slowly or not at all. As a result, the network learns slowly or not at all. ResNet solves this by introducing residual connections, which allow gradients to bypass certain layers (He et al., 2016). This maintains stronger gradients and improves training efficiency as a result.

RNN. RNNs are specialised in handling sequential data, where the temporal order of information is important. Unlike feedforward networks, such as CNNs, RNNs maintain hidden states that

carry information from previous time steps. This makes them suitable for tasks like Natural Language Processing (NLP) and time series analysis. Two key variants of RNNs are the GRU (Cho, 2014) and LSTM (Hochreiter, 1997), which were designed to overcome the limitations of standard RNNs, particularly the vanishing gradient problem (Sarker, 2021a). Both GRU and LSTM operate based on gates that regulate the flow of information, allowing these architectures to capture long-term dependencies in sequence data. The key difference between GRU and LSTM is the number of gates, where the former has two and the latter three. Consequently, LSTMs might be suitable for learning more complex patterns at the price of computational complexity. A general schematic of an LSTM building block is shown in figure 2.3.

GAN. GAN (Goodfellow et al., 2014) is a neural network architecture designed for generative modelling, which focuses on producing new, realistic samples from a given dataset. Simply, it captures the patterns within the input data, which makes it possible to generate new examples that closely resemble the original data. GANs are widely used in tasks such as image generation, creative applications, and data augmentation. GANs consist of two competing networks, a generator and a discriminator, working in tandem (Sarker, 2021a). The generator produces synthetic data, while the discriminator evaluates its authenticity, pushing the generator to create increasingly realistic outputs.

Autoencoder. AE is an unsupervised learning architecture designed for dimensionality reduction, data compression, and feature extraction (Goodfellow et al., 2016). They consist of an encoder that compresses the input data and a decoder that reconstructs it from this compressed representation, making them useful for tasks like noise reduction and anomaly detection. AEs are also prominent in generative modelling, now standing alongside other methods such as GANs.

Transformer. The transformer architecture (Vaswani, 2017) utilises self-attention mechanisms, enabling each element in an input sequence to attend to all other elements simultaneously. While initially developed for machine translation, the transformer allows for efficient capture of long-range dependencies on a general level, which made it fundamental to many other NLP tasks. Specifically, its ability to model complex relationships without relying on recurrent units has made it the foundation for large-scale models such as GPT (Liu et al., 2023) and BERT (Devlin, 2018). ViT is a specialised application of the same principle to computer vision (Dosovitskiy, 2020). In standard ViT, the self-attention mechanism is used to process image data by treating images as sequences of patches, demonstrating competitive performance with traditional CNNs in many tasks.

2.2.3 Methods

While the field of DL is extensive, we focus on a selection of prominent methods. These methods represent key areas of ongoing investigation and highlight the diversity of approaches driving innovation and advancement in the domain. These methods are: (1) Attention, (2) Few-shot learning, (3) Continual learning, (4) SSL, (5) Active learning (AL), (6) Neural Architecture Search (NAS), (7) Meta-learning, (8) Multimodal learning and (9) Federated Learning. We cover each of these in turn.

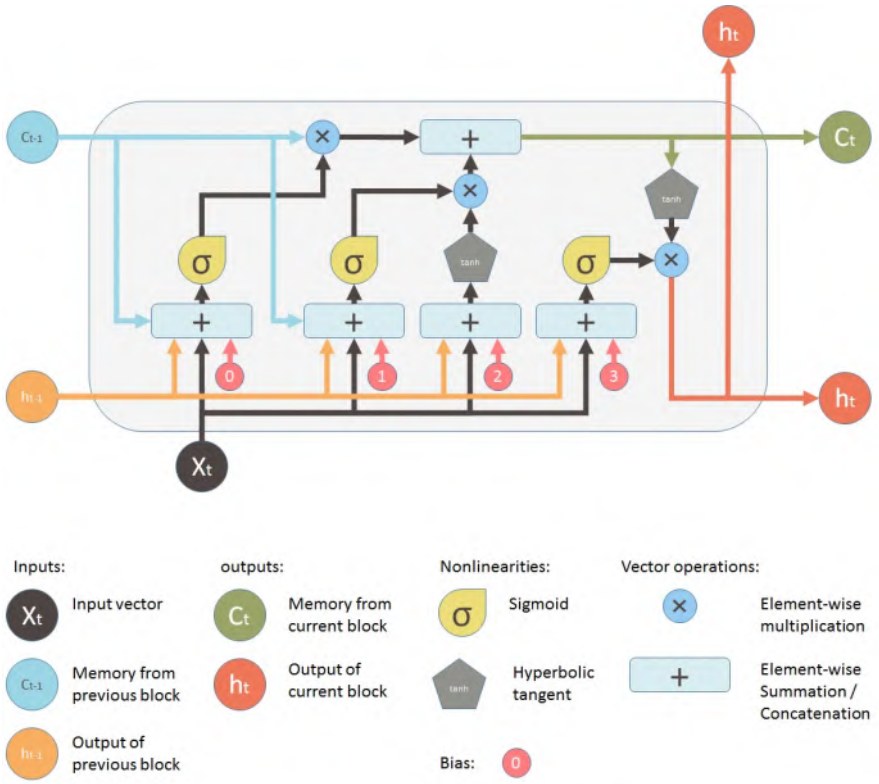


Figure 2.3: LSTM building block. Image source: (Shi, 2016)

Attention. Attention mechanisms allow models to assign different importance to various parts of the input data, focusing on the most relevant elements (Guo et al., 2022). This is particularly effective in sequence-based tasks such as NLP, where certain words or tokens carry more weight in determining the context and meaning of a sentence. In computer vision, attention mechanisms have also been applied to images, where they help models focus on key regions of an image, improving performance by identifying the most relevant spatial features.

Few-shot Learning. Few-shot learning aims to train models that can generalise from only a few examples per class (Song et al., 2023). This method is particularly useful in scenarios where collecting a large amount of annotated data is not feasible or possible, leveraging prior knowledge to perform well with minimal training data (Xue et al., 2024).

Continual Learning. Continual learning, also known as incremental or lifelong learning, involves training models on dynamic data distributions where new tasks, skills, or environments are introduced over time (Wang et al., 2024). Unlike traditional models that assume static data, continual learning adapts to changing inputs. A key challenge is catastrophic forgetting, where learning new information can diminish the model's ability to retain previous knowledge. This challenge reflects the balance between plasticity (adapting to new tasks) and stability (preserving learned knowledge).

SSL. SSL leverages unlabelled data by creating auxiliary tasks, enabling the model to learn useful representations without requiring explicit labels (Gui et al., 2024). In an auxiliary or pretext

task, the pseudo-labels are automatically generated based on inherent data properties. Simply, the data itself is used to supervise learning with the loss function designed depending on the configuration of the pretext task. The pretext loss function might be also used for the final task - for example, as in depth estimation (Garg et al., 2016). The model is trained on these tasks during a pre-training phase and then fine-tuned for downstream applications. A key advantage of SSL is its ability to leverage large-scale unlabelled data, reducing reliance on costly human annotations.

AL. AL seeks to reduce the amount of annotated data required for training by selectively identifying the most informative data points for annotation (Tharwat and Schenck, 2023). The underlying assumption is that training on intelligently chosen, representative examples could achieve similar performance to using indiscriminately annotated datasets, while significantly lowering annotation costs. This method is especially useful when annotation is expensive or time-consuming, as typically only the most uncertain or ambiguous samples are actively queried for annotation. A schematic of a general deep AL method is shown in Figure 2.4.

NAS. NAS automates the design of neural network architectures by searching over a space of possible architectures to find the one that optimises performance on a given task (Salehin et al., 2024). This process reduces the need for human expertise in crafting neural networks, potentially leading to architectures that outperform manually designed models.

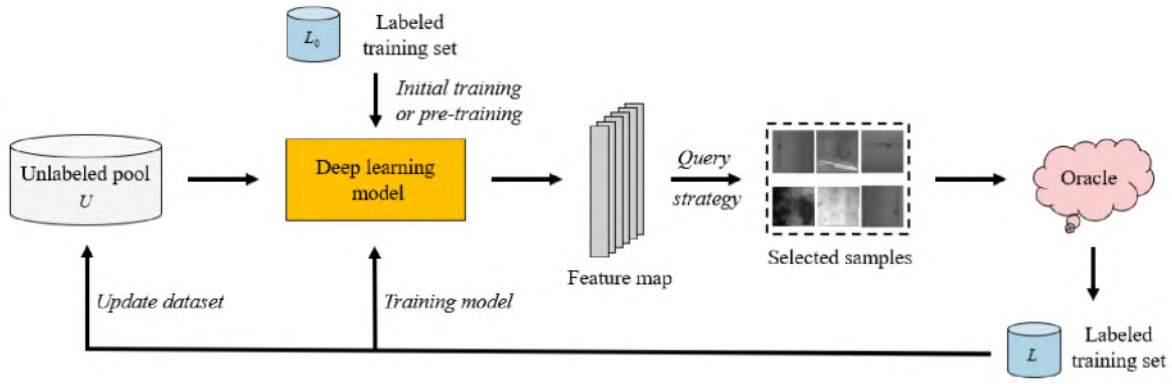


Figure 2.4: A general schematic of the deep AL pipeline, where the oracle, typically a human annotator, provides labels for the selected samples. Image source: (Ren et al., 2020)

Meta-Learning. Meta-learning, often described as 'learning to learn', enables models to adapt quickly to new tasks by learning how to optimise the learning process itself (Vettoruzzo et al., 2024). Meta-learning enables models to optimise their own learning processes, mimicking the way humans generalise from prior experience to acquire new skills. Specifically, instead of learning task-specific representations, meta-learning focuses on learning transferable strategies that facilitate efficient adaptation. Meta-learning is valuable given the large amounts of data typically required to train models from scratch.

Multimodal Learning. Multimodal learning integrates information from multiple data modalities, such as text, images, audio and potentially others into a single model (Jabeen et al., 2023). Simply, by combining data from multiple modalities, multimodal learning uncovers features that might remain hidden in single-modality approaches. Specifically, complementary information

from another modality was shown to improve overall model performance. Multimodal learning enables numerous applications, including audio-visual speech recognition, multimedia indexing and retrieval and healthcare analysis (Jabeen et al., 2023).

Federated Learning. Federated learning decentralises model training by distributing the learning process across multiple devices, where each device computes updates locally without sharing raw data (Wen et al., 2023). This method is crucial for preserving user privacy and complying with data protection regulations, particularly in sensitive fields like healthcare and finance. Additionally, federated learning reduces the server's computational and storage load by performing training on local devices. By aggregating updates from these distributed devices, federated learning can produce a global model that outperforms models trained on individual devices, all without centralising sensitive information.

2.2.4 Datasets

Having large, well-curated datasets is essential for many DL tasks. These datasets can be either publicly available or privately held, depending on factors such as domain, legal regulations, and data sensitivity. General-purpose datasets, such as ImageNet (Deng et al., 2009), which consist of non-sensitive and widely available natural data, are typically easier to collect and access. In contrast, datasets from more specialised domains, like medical datasets, are more challenging to obtain, as discussed in section 1.5.3. Overall, large dataset creation poses significant challenges in such aspects as data annotation, data curation, and data privacy. This is exacerbated for especially large datasets, crucial for training foundation models, which require vast amounts of data to support a wide range of applications and often achieve higher accuracy than smaller models. Furthermore, as shown in figure 2.5, year on year improvement of DL performance on classical datasets, such as Cityscapes (Cordts et al., 2016), COCO (Lin et al., 2014a) and ImageNet, has reduced during the recent years. This could suggest either a plateau in AI capabilities – despite the increasing success of larger models across many domains – or the need for larger, more diverse, and realistic datasets that accurately represent their domains for both learning and benchmarking. However, creating such datasets remains a significant challenge.

Considering the size of the DL domain, and the large number of datasets being released, we cover some of the most recent and impactful publicly available datasets generally relevant to this work. Specifically, 4 datasets in the general computer vision category: (1) Open Images, (2) Large Vocabulary Instance Segmentation (LVIS), (3) SA-V and (4) Objects365. These are followed by 5 datasets in medical imaging category: (5) TotalSegmentator dataset, (6) National Lung Screening Trial (NLST), (7) AbdomenAtlas, (8) MedShapeNet, (9) Uterine Myoma MRI dataset (UMD).

We present the four general computer vision datasets in turn.

Open Images V7. Open Images is a large-scale dataset of 9 million natural images. It contains 16 million bounding boxes across 600 object classes, 2.8 million instance segmentations on 350 classes, and 61.4 million image-level labels for 20,638 classes. This dataset also includes visual relationship and localised narrative annotations, making it suitable for a wide range of computer vision applications (Benenson and Ferrari, 2022).

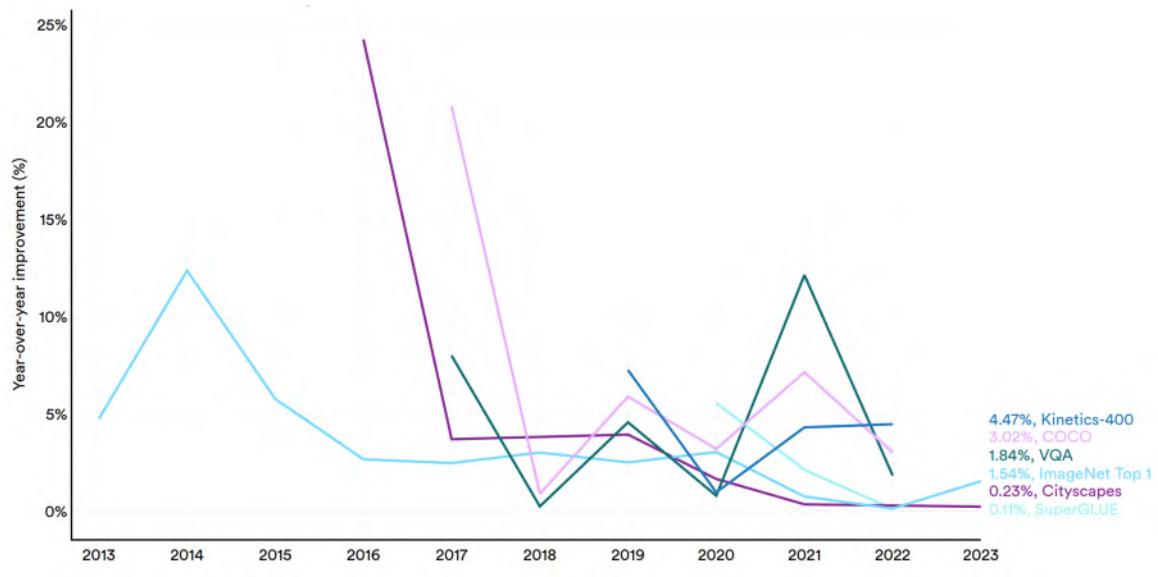


Figure 2.5: Year-over-year improvement of DL performance on classical datasets. Source: (Maslej et al., 2024)

LVIS. LVIS (Gupta et al., 2019) includes 164,000 natural images with approximately 2 million crowdsourced instance segmentation annotations for over 1,200 categories, focusing on long-tail object distributions. The dataset emphasizes rare objects, making it valuable for tasks requiring detailed segmentation across a wide range of categories. It is complementary to COCO, another large natural image dataset where the segmentations are generally coarse, by providing precise instance-level annotations for the 164,000 images in COCO 2017.

SA-V. SA-V is a large-scale natural video dataset designed for training and evaluating general-purpose object segmentation models. It comprises 51.000 diverse, high-resolution videos with 643.000 spatio-temporal segmentation masks. The videos cover a wide range of subjects, including locations, objects, and scenes, with class-agnostic masks ranging from large structures like buildings to fine details such as interior decorations. Annotations are done both manually and with assistance of SAM 2.

Objects365. Dataset Objects365 is a large-scale object detection dataset comprising 2 million images annotated with more than 30 million bounding boxes across 365 object categories. These categories are derived from 11 supercategories that represent common objects in daily life. The dataset includes categories from PASCAL VOC (Everingham et al., 2015) and COCO (Lin et al., 2014a) benchmarks for compatibility, providing diverse and densely annotated images to facilitate training and evaluation of object detection models.

We present the five medical imaging datasets in turn.

TotalSegmentator dataset. TotalSegmentator is an open-source tool developed for automatic segmentation of medical imaging data. It was trained on 1,228 CT volumes to segment 104 anatomical structures and 298 MRI volumes to segment 59 structures. The training data is made available as a dataset (Wasserthal et al., 2023; D'Antonoli et al., 2024).

NLST. NLST is a segmentation dataset focused on lung anatomy. The data is sourced from the NLST and automatically annotated using the TotalSegmentator tool (Thiriveedhi et al., 2024). It comprises 126,088 CT volumes with 9,565,554 anatomical structures annotated in total. While this means that annotations are essentially weak, the sheer size of this dataset makes it very useful in the data-starved medical image analysis domain.

AbdomenAtlas. AbdomenAtlas 1.1 contains 9,262 CT volumes with voxel-wise annotations for 25 organs and pseudo-annotations for seven types of tumours produced manually and semi-automatically by a team of 10 radiologists (Li et al., 2024). Such dataset size enables creation of the large pre-trained models in the medical imaging domain, while providing a comprehensive benchmark for evaluating other methods.

MedShapeNet. MedShapeNet (Li et al., 2023) is a large-scale collection of over 100,000 3D medical shapes derived from imaging data of real patients, including healthy and pathological subjects. It encompasses a wide range of anatomical structures —including bones, organs, and vessels— as well as 3D models of surgical instruments. The dataset is compiled from 23 different datasets, with each shape paired with ground truth annotations.

UMD. UMD is the largest publicly available uterine MRI segmentation dataset to date, consisting of 300 cases of uterine myoma T2-weighted sagittal images (Pan et al., 2024). It encompasses 9 types of uterine myomas classified by the International Federation of Gynaecology and Obstetrics (FIGO), with annotations reviewed by 11 experienced doctors. UMD is a valuable resource for clinical research, since annotated segmentations of female abdominal MRIs are very scarcely available in public.

Figure 2.6: Improvement of image generation results in Midjourney in the span of 2 years between February, 2022 and December, 2023 with a query 'a hyper-realistic image of Harry Potter'. Source: (Maslej et al., 2024)

2.2.5 Recent Milestones & Tendencies

In recent years, DL has rapidly advanced due to increased computational power, availability of large datasets, and new model architectures. Certain advances exemplify this trend. AI systems have surpassed human performance in several tasks, including image classification since 2015, basic reading comprehension since 2017, visual reasoning since 2020, and natural language inference since 2021 (Maslej et al., 2024), as shown in figure 2.7. In mathematical problem-solving, per-

formance on the MATH dataset, comprising 12,500 competition-level problems, improved from 6.9% in 2021 to 84.3% in 2023 using Generative Pre-trained Transformer 4 (GPT-4)-based models (Liu et al., 2023). Generative models like Midjourney (Midjourney, 2022) have shown remarkable progress in creating hyper-realistic images over two years, as shown in figure 2.6. In code generation, AI systems have significantly advanced on the HumanEval benchmark (Chen et al., 2021a), with GPT-4 variants achieving a 96.3% success rate with an increase of 64.1 percentage points since 2021. Advances in audio generation were marked by the release of models like MusicGen (Copet et al., 2024), and MusicLM (Copet et al., 2024), improving the synthesis of human speech and music.

Key trends driving these advancements include the development of foundation and pre-trained models, demonstrating high performance and generalisation capabilities. Furthermore, multi-modality has enabled models to process and generate data across different domains, leading to breakthroughs in generative models such as Stable Diffusion (Podell et al., 2023), Midjourney and ChatGPT (Liu et al., 2023). Specifically, LLMs have demonstrated significant capabilities in natural language understanding and generation, often adopting attention mechanisms. However, AI still lags in complex cognitive areas like visual commonsense reasoning and competition-level mathematical problem-solving, and there is a growing emphasis on explainable AI to enhance the interpretability and trustworthiness of DL models in critical applications. We cover these trends in turn: (1) Pre-trained & Foundation Models, (2) Multi-modality, (3) Generative Models, (4) Transformers, and (5) Explainable AI (XAI).

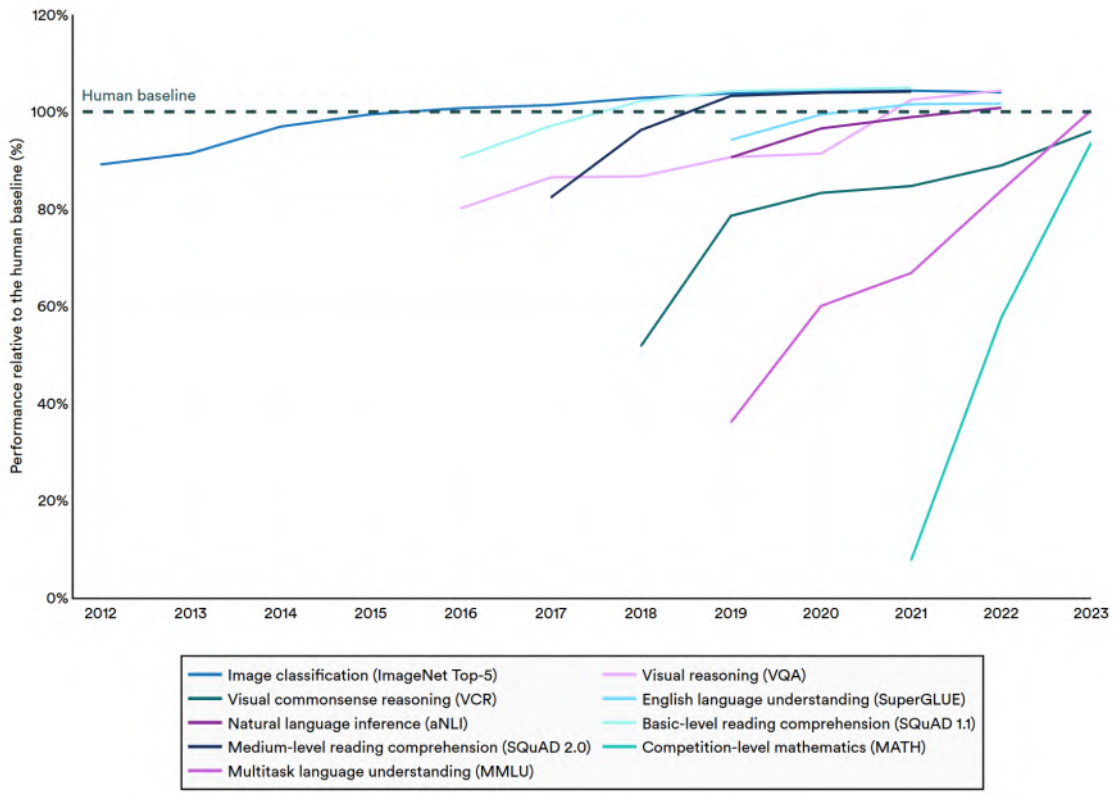


Figure 2.7: Select AI performance benchmarks vs. human performance. Source: (Maslej et al., 2024)

Pre-trained & Foundation Models. A pre-trained model is one that has already been trained on a dataset and can be shared, used, and modified if necessary. With the rise of DL, these models have

become widely distributed and utilised, even by non-experts, especially due to the growing public interest in open-source image generation tools such as Stable Diffusion. The availability of these models provides a starting point for various tasks, allowing them to be directly used or employed for creating new models.

Foundation models, such as Stable Diffusion, SAM, and TotalSegmentor, are a subset of pre-trained models trained on especially large datasets and often designed to generalize across multiple tasks. Trained on vast and diverse datasets, these models can be adapted to new applications through methods such as fine-tuning or model distillation, without requiring complete retraining. Recently, they have been produced for numerous tasks, including image, video, and audio generation, NLP, object detection, image segmentation and classification. The quantity of produced foundation models has doubled from 72 to 149 in years 2023 and 2024 respectively (Maslej et al., 2024) as shown in figure 2.8. However, producing such models is often highly expensive and resource-intensive. For instance, Gemini Ultra (Team et al., 2023), a foundation model developed by Google, costs $191.4 million to train (Maslej et al., 2024). The high cost and large data requirements make it impractical or impossible to develop such models in certain domains. Specifically, in the context of foundation models for medical imaging, the large data volumes required to train typical non-medical foundation models were not yet reached (Zhou et al., 2023a).

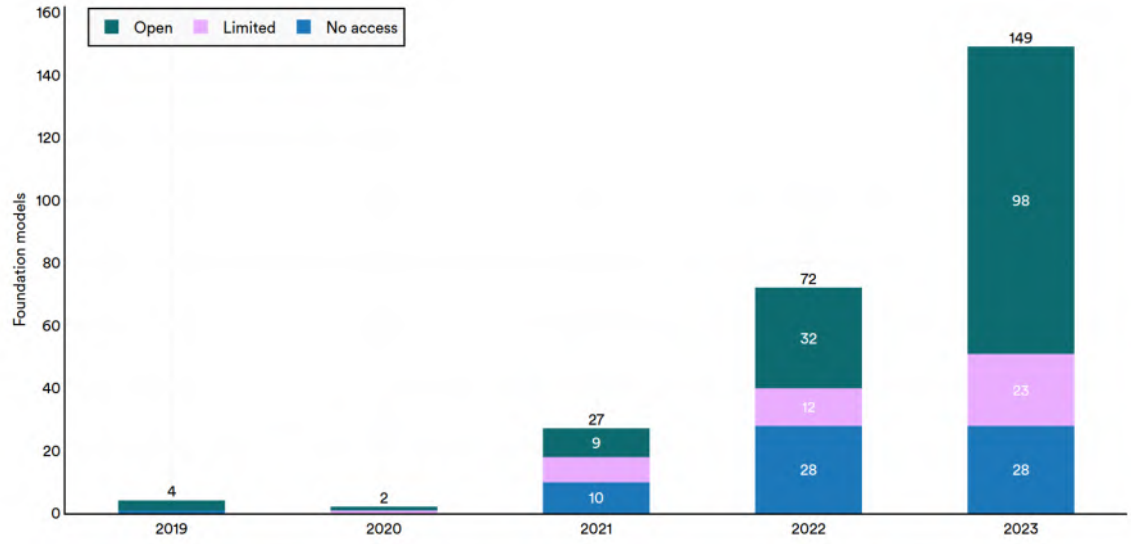


Figure 2.8: Foundation models by access type. Source: (Maslej et al., 2024)

Multimodality. Multimodality refers to models that integrate and process multiple types of data inputs—such as text, images, audio, and video—allowing them to handle various data formats simultaneously. Recent advancements have led to powerful multimodal models like Contrastive Language-Image Pre-training (CLIP) (Radford et al., 2021), DALL-E (Ramesh et al., 2021), Gemini, and GPT-4, which can generate images from text descriptions or generate text from speech. In the medical field, multimodal approaches combine data types like medical images, biosignals, clinical records, and other relevant sources to achieve a more comprehensive understanding of patient conditions (Yin et al., 2023). For example, in Alzheimer's disease diagnosis, models using only structural MRI scans or speech analysis achieve approximately 80% detection accuracy. By incorporating additional modalities such as audio features, speech transcripts, genomic data, and clinical assessments, multimodal models have improved diagnostic accuracy to over 90% (Salvi et al., 2023).

Generative Models. Generative models can produce synthetic data closely resembling real-world data. Recent advancements with such models, including OpenAI's GPT, Stable Diffusion, and MidJourney, have led to the creation of text, images, and even music, which become increasingly difficult to distinguish from real data (Bandi et al., 2023). This ability has significant implications across multiple industries by enabling applications in domains where data is scarce or difficult to obtain. For example, for rare conditions in healthcare. However, reliance on synthetic data presents challenges: models trained predominantly on synthetic data can experience model collapse. This is a condition, when the model loses the ability to represent true data distributions and produces less diverse outputs. Furthermore, statistical evaluations show that synthetic data often has higher dissimilarity to real data and reduced quality and diversity. While incorporating real data through techniques like synthetic augmentation loops can mitigate some issues, both fully synthetic and augmented methods exhibit diminishing returns with continued training (Shumailov et al., 2023).

Transformers. Transformer architectures are increasingly instead or in combination with CNNs in many domains, especially for tasks involving sequences and long-range dependencies. Unlike CNNs, which utilise local receptive fields and stationary convolutional filters, transformers can adaptively focus on different parts of the input data. In vision tasks, ViTs are now on par or outperforming CNNs in various benchmarks (Shamshad et al., 2023), including semantic segmentation (Zheng et al., 2021). While Transformers are more computationally demanding, they mitigate the inductive biases typical of CNNs, which leads to their growing usage in tasks that benefit from the model learning complex spatial relationships and global features, such as medical imaging.

Explainable AI. XAI seeks to improve the explainability of AI, by making the inner workings of complex ML systems, particularly DL models, understandable. This is done by providing insights into how the predictions are produced. XAI has become essential in ensuring transparency and trust in DL models, particularly in sensitive fields like healthcare, finance, and legal systems (Dwivedi et al., 2023). Specifically, In healthcare clinical decisions depend on both accuracy and interpretability, and high performance model's metrics alone are insufficient (Chaddad et al., 2023). This is addressed by notable XAI methods, including Shapley Additive Explanations (SHAP) (Lundberg, 2017), Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016), Class Activation Mapping (CAM) (Zhou et al., 2016), Grad-CAM (Selvaraju et al., 2017) and their variations. Despite these advancements, XAI still cannot sufficiently enhance the interaction between users and AI models due to its inability to provide clear, interpretable explanations and strong evidence. This limitation poses challenges in meeting the expectations of medical experts, who tend to disregard these explanations as a result (Chaddad et al., 2023).

2.2.6 Libraries

In any programming language, a library is a collection of prewritten code that developers can use to perform common tasks, thereby avoiding the need to write code from scratch. Libraries provide standardised solutions and functions, which expedite development. This concept is especially sig-

nificant in DL for two reasons: (1) DL involves computationally complex operations that require extensive optimization to run efficiently, and (2) DL is a rapidly evolving field with numerous algorithmic components originating from different domains. Therefore, using libraries allows developers to streamline employing complex optimizations and integrating diverse algorithms without reinventing the wheel (Raschka et al., 2020; Tufail et al., 2023).

Building on this, Python (Srinath, 2017) has emerged as one of the leading programming languages in DL due to its readability, simplicity and ecosystem. Its high-level syntax and extensive ecosystem of scientific computing libraries allow for rapid prototyping and development (Raschka et al., 2020). To balance ease of use with computational efficiency, many Python libraries are built on lower-level languages like C++ or CUDA (Luebke, 2008). This approach leverages Python's user-friendly interface while harnessing the performance benefits of statically typed languages.

Python's libraries in DL can be broadly categorised into two types: (1) end-to-end ML libraries, and (2) specialised libraries that provide specific functionalities. End-to-end libraries, such as PyTorch (Paszke et al., 2019), TensorFlow (Abadi et al., 2015), and Keras (Chollet et al., 2015), offer comprehensive frameworks for building, training, and deploying neural networks. They enable developers to manage all aspects of the DL pipeline within a unified environment. In contrast, specialised libraries like NumPy (Harris et al., 2020) for numerical computations, Matplotlib (Hunter, 2007) for data visualisation, and NiBabel (Brett et al., 2024) for working neuroimaging data formats offer specific functionalities that often support DL tasks, but do not encompass the entire ML workflow.

While the DL ecosystem is extensive and the choice of an end-to-end library is largely task-dependent, we focus on several prominent specialised libraries that are instrumental in DL generally, and in computer vision and medical imaging specifically. These libraries are: (1) MRQy, (2) Pytorch Image Models (timm) & Segmentation Models, (3) Medical Open Network for AI (MONAI) and (4) MLflow & aim as examples of Machine Learning Operations (MLOps) libraries. We cover each of these in turn.

MRQy. MRQy (Sadri et al., 2020) is an open-source quality control tool designed to quantitatively assess and compare MRI and CT series within and between large imaging cohorts. Its purpose is to identify differences arising from factors like site-specific or scanner-specific variations (e.g., image resolution, field-of-view, contrast settings) and imaging artefacts such as noise or motion. By extracting quality measures and metadata from the underlying data, MRQy helps detect these variations before further use. This is especially valuable, since in medical imaging data variability arises from numerous factors and can negatively affect resulting models: for example, by introducing bias or limiting accuracy and generalizability.

timm & Segmentation Models. timm (Wightman, 2019) and Segmentation Models (Iakubovskii, 2019) libraries provide access to numerous pre-trained deep learning models for computer vision. timm offers over 700 state-of-the-art models, primarily trained on ImageNet, while Segmentation Models focuses on pre-trained weights for image segmentation with 124 models included. Other key sources of pre-trained models include Hugging Face, TensorFlow Hub, PyTorch Hub and OpenAI's Model Zoo. These resources expedite deep learning prototyping, research, and deployment by enabling the use of advanced models in a plug-and-play fashion without the need for training from scratch, which is resource-intensive.

MONAI. MONAI (Diaz-Pinto et al., 2022) is an open-source framework built on PyTorch, specifically designed for DL applications in healthcare imaging. It offers a suite of tools and functionalities that address the unique requirements of medical image analysis, such as handling multi-dimensional data and incorporating domain-specific transformations. Specifically, MONAI provides an end-to-end pipeline for developing medical imaging models. It covers stages from data input and preprocessing to model training, evaluation, and deployment. Medical imaging algorithms are often under-represented in general DL libraries. MONAI fills this gap by offering specialised components tailored to the specifics of medical data.

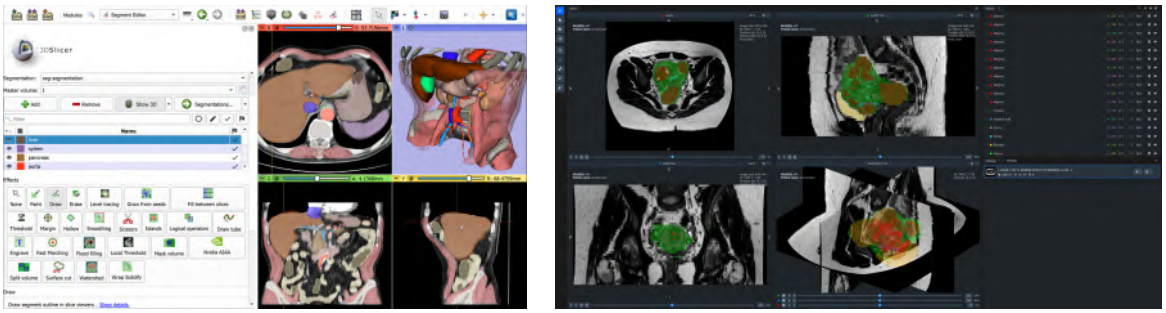
MLOps: MLflow & aim. MLOps refers to the set of practices and tools that streamline the development, deployment, monitoring, and maintenance of ML models in production environments. Its importance lies in enabling a collaborative and efficient transition from model development to deployment and ongoing maintenance, while ensuring consistent model performance and reproducibility. MLflow (Zaharia et al., 2018) and aim (Arakelyan et al., 2024) are two notable open-source MLOps libraries that facilitate different aspects of the ML lifecycle. While both tools support experiment tracking, MLflow excels in model versioning and deployment capabilities, making it suitable for projects that require robust model management. Aim is preferred for its deeper experiments tracking, visualisation and analysis instruments. Alternative MLOps tools include Neptune.ai, Weights & Biases, TensorBoard, Metaflow and Vertex AI, and often focus on different MLOps aspects.

2.2.7 Computational Resources

On hardware level, the calculations in DL are typically performed by one or multiple GPUs. A GPU is a specialised hardware component designed to perform rapid mathematical computations in parallel. Specifically, GPUs enable efficient training of neural networks by handling multiple calculations simultaneously. Advancements in DL architectures, such as transformers like GPT or iterative improvement of generative models like Stable Diffusion, have shown that model performance often correlates with model size. For instance, between 2014 and 2017, the size of winning models in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al., 2014) grew significantly, from approximately 4 million parameters in 2014 to 146 million in 2017. In contrast, GPU memory capacity only tripled during the same period (Raschka et al., 2020), which resulted in a bottleneck. Simply, to achieve better performance one needs a deeper model with more layers, which might not be loaded on a single GPU. To address this challenge, various strategies might be employed. While making DL models smaller and faster is an active research domain, model parallelisation permits fitting a model across multiple GPUs. This allows training larger models that exceed the memory capacity of a single GPU, provided a GPU cluster is available.

A GPU cluster is a networked group of interconnected computers (nodes), where each node is equipped with one or more GPUs. Cluster sizes can vary significantly, ranging from just a few GPUs to tens of thousands (Meta, 2024). To efficiently manage and optimise the use of GPU clusters in training large-scale DL models, workload managers or job schedulers such as the Simple Linux Utility for Resource Management (Slurm) (Jette and Wickberg, 2023) are commonly employed.

Slurm, in particular, is an open-source workload manager used in high-performance computing, responsible for job scheduling and resource allocation by distributing computational tasks across cluster nodes. The deployment of such clusters can be facilitated by tools like DeepOps (Majee, 2024), which streamlines rollout of the cluster management software. In this work, we utilised two clusters: (1) the Mésocentre Clermont-Auvergne cluster at Université Clermont Auvergne, equipped with up to 8 GPUs of various models, and (2) an in-house cluster with 2 GPUs, which was set up by us using DeepOps. Unless otherwise stated, all reported results were obtained on the latter.



(a) 3D Slicer (b) Supervisely

Figure 2.9: Annotation workspace GUI: 3D Slicer and Supervisely.

2.2.8 Software

As DL models and datasets increase in size and complexity, specialised software becomes all the more essential. DL research, development, deployment, and maintenance are supported by a wide array of software specialised software tools, that are often task-specific. These tools can be open-source or proprietary. Here, we focus on tools relevant to computer vision and, more specifically, to medical imaging and segmentation. Among the most frequently used software types in these domains are data annotation and data analysis tools. The notable tools include: (1) 3D Slicer, (2) Medical Imaging Interaction Toolkit (MITK), (3) Supervisely, (4) Synapse 3D, and (5) MeshLab. We discuss each of these tools in turn.

3D Slicer (Kikinis et al., 2013), MITK (Goch et al., 2017), Supervisely (Supervisely OU, 2024), and Synapse 3D (Fujifilm, 2024) are advanced tools for medical image analysis, including segmentation. They share a common focus, but differ in the range of functionalities offered. 3D Slicer is one of the most feature-complete tools and offers modules for visualisation, processing, segmentation, registration, and analysis of medical and biomedical images, supported by a collaborative community. In turn, MITK provides a simpler, less exhaustive framework for viewing, processing, and segmenting medical images. In contrast to both 3D Slicer and MITK, Supervisely is an integrated ecosystem, which emphasises collaborative data annotation, data management and model training. For reference, the GUIs of both 3D Slicer and Supervisely are shown in figure 2.9. Synapse 3D, developed by Fujifilm, automates three-dimensional image segmentation using proprietary image intelligence technology, enhancing accuracy in radiological and surgical workflows. MeshLab (Cignoni et al., 2008) stands apart from these tools as it specialises in the processing and editing of 3D triangular meshes. While MeshLab is not exclusively designed for medical imaging, it might be used for preparing and refining 3D models derived from medical scans. With the exception of Supervisely and Synapse 3D, all tools are open-source.

2.3 Machine Learning in Clinical Practice

The contributions of this work lie within the intersection of ML and clinical practice. To provide context, we review the specifics of ML in clinical practice, which is actively researched and increasingly applied for assistance in a variety of tasks. This includes several key areas: (1) diagnostics, (2) treatment, (3) population health management and (4) patient care (Alowais et al., 2023; Bahl, 2022; Nwanosike et al., 2022). In diagnostics, researchers are exploring how ML can enhance diagnostic accuracy and integrate genomic medicine. For treatment, ML is being investigated to support precision medicine and optimise dosing and therapeutic drug monitoring. In population health management, applications are studied for predictive analytics, risk assessment, and providing drug information and consultation. Additionally, AI-powered patient care is a promising area, including virtual healthcare assistance and mental health support (Alowais et al., 2023). Despite the potential of these advancements, several challenges impede the adoption of ML in clinical decision-making. These challenges can be generally split into two groups, depending on their source, making a total of eight challenges. First, there are five challenges stemming from inherent specifics of ML: (1) evaluation, (2) reproducibility and generalizability, (3) explainability, (4) usability and (5) security. Second, there are three challenges stemming from unique demands of clinical practice: (6) limited data availability, (7) data quality, variability, and (8) ethical and legal considerations (Hofer et al., 2020; Daye et al., 2022). We cover each of these two groups in turn.

Figure 2.10: Percentage of respondents reporting risks associated with generative AI that negatively impacted organizations. Based on responses from organizations using generative AI in at least 1 function. Total number of respondents is $n = 876$. Responders, who chose 'don't know/not applicable' (17%), are not shown. Source: (Singla et al., 2024)

2.3.1 ML Challenges

ML is celebrated for its ability to reduce costs, enhance efficiency, and improve accuracy and precision in a number of tasks. However, inherent challenges related to reliability, explainability and security hinder its widespread adoption. Figure 2.10 illustrates the risks that caused negative consequences organisations faced in early 2024 when implementing generative models. The top three risks are inaccuracies (23%), security vulnerabilities (16%), and explainability shortcomings (12%). Although generative models represent only a subset of ML, the concerns about the reliability of ML models in general are well-documented in the literature (Nwanosike et al., 2022; Alowais et al., 2023; Bahl, 2022). We cover each of the five key ML challenges in turn.

Evaluation. Evaluating ML models in clinical practice presents significant challenges due to the high stakes involved in patient care. Therefore, the robustness and technical readiness of an ML model must be thoroughly assessed before it can be safely integrated into clinical workflows. However, standard metrics and evaluation practices are not exhaustive, and may not adequately capture the model's performance in real-world clinical settings. While there is no standardised evaluation protocol, several guidelines were proposed (Daye et al., 2022). They include the Standards for Reporting Diagnostic Accuracy Studies (STARD-AI) (Sounderajah et al., 2020), the Transparent Reporting of a Multivariable Prediction Model for Individual Prognosis or Diagnosis (TRIPOD-AI) (Collins and Moons, 2019), and the Consolidated Standards of Reporting Trials (CONSORT-AI) (Liu et al., 2020).

Reproducibility and generalizability. To integrate ML models into clinical practice, their outputs should satisfy two key criteria: reproducibility and generalizability. Simply, ML models must maintain their predictive performance not only on the training data, but also across diverse patient populations and clinical settings. This requires that the datasets used for model development accurately reflect the environments where they will be applied. Furthermore, the ML model and its training process must be able to capture this information. Careful assessment of what is included or excluded from the dataset, and evaluation of its effects are essential to identify and mitigate bias (Daye et al., 2022).

Explainability. Many ML models, especially DL models, operate as 'black boxes' that provide outputs without a transparent comprehensible reasoning process behind (Challen et al., 2019). This lack of interpretability makes it difficult for medical experts to understand or verify how decisions are made, which hinders ML application in clinical practice. Furthermore, if clinicians cannot explain how a diagnosis, treatment plan, or prognosis was determined, it can undermine patient trust, as patients expect clear justifications for medical decisions (Sim et al., 2023).

Usability. Given the high stakes of clinical decisions, along with legal and regulatory concerns, it is essential that ML applications are designed for interactive use. This ensures that human experts remain in control, allowing them to validate or adjust model outputs based on their expertise. Usability, defined as the ease with which users can interact with a system to achieve their goals, is critical in this context. In medical environments, where decisions are high-stakes, complex or non-intuitive interfaces, as well as difficult-to-interpret results, can lead to errors, hinder the adoption of ML applications, and discourage clinicians from using them (Daye et al., 2022).

Security. Security poses a significant challenge in the application of ML models in clinical practice, largely due to their vulnerability to input perturbations. Adversarial attacks (Puttagunta et al., 2023) exploit this weakness by intentionally manipulating inputs to induce errors in model outputs. These alterations might often be minor and imperceptible to human observers. However, they might lead these models to produce incorrect or misleading predictions, which may have severe consequences in healthcare settings, such as misdiagnosis or inappropriate treatment recommendations. Therefore, security in the context of ML models involves not only protecting access to sensitive patient data, but also safeguarding the integrity of the models and their outputs (Daye et al., 2022).

2.3.2 Clinical Practice Challenges

While the establishment of AI infrastructure in clinical practice has been extensively discussed in the literature (Willemink et al., 2020b; Jha and Topol, 2016; Karalis, 2024), these discussions are often written from the engineering or data science perspective. In particular, they often emphasise technical components such as data access, security, cross-platform integration, and algorithm development, overlooking the unique challenges inherent in the clinical environment. As a result, critical issues related to limited data availability, data quality and variability, and ethical and legal considerations specific to clinical practice are often not fully addressed (Daye et al., 2022). We cover each of the 3 key clinical practice challenges in turn.

Limited data availability. Each year, an estimated 50 petabytes of medical data are generated, which includes clinical notes, laboratory results, and medical images. Despite the vastness of these data, 97% of it remains unanalyzed and underutilised (Dutta et al., 2019). Several factors contribute to this underutilization, including stringent data privacy regulations, lack of standardisation and the fragmentation of data across different departments, devices, and institutions. For these reasons, publicly available medical data is severely limited and impedes progress in this field (Adlung et al., 2021). These issues are especially pronounced in specific domains, such as in FPMRI segmentation, that mostly rely on private small datasets and have not been extensively researched. Rare diseases face an even more critical challenge due to the inherent scarcity of documented cases.

Data quality and variability. Data quality and variability pose significant challenges to the application of ML in clinical practice. Medical data are inherently heterogeneous and complex both across the medical domain as a whole, due to diverse data sources, and within specific domains, reflecting variability between patients and technologies employed. This leads to inconsistencies and ambiguities, especially given the evolving nature of medicine in general, its terminology and technologies employed. Poor-quality and skewed data can introduce biases and errors into ML models, resulting in unreliable or erroneous conclusions. Issues can arise at various stages, including data collection, coding, and standardisation, influenced by technical and organisational factors. Evaluating and controlling the quality and variability of clinical data is essential for developing trustworthy models, which is an active research domain (Bernardi et al., 2023).

Ethical and legal considerations. Ethical and legal considerations pose significant challenges to integrating ML into clinical practice. The reliance of ML algorithms on large datasets raises concerns about patient privacy and data security. For example, inadequate data safeguarding can lead to serious consequences, as seen in the case where the Royal Free London Trust transferred patient information to DeepMind without consent, resulting in legal repercussions (Sim et al., 2023). The dynamic nature of ML algorithms also complicates regulatory compliance and certification processes across different regions. To address these issues, there is a growing need for comprehensive regulations. The Artificial Intelligence Act (The European Parliament and the Council of the European Union, 2024b) is an example of efforts to establish a regulatory framework that ensures transparency, accountability, and ethical use of AI technologies in healthcare. Despite ongoing efforts,

the fast-evolving nature of the AI domain makes it complicated to produce clear and straightforward regulations, which results in regulatory hurdles during adoption of ML models in clinical practice (Adlung et al., 2021; Daye et al., 2022).

Chapter 3

Data

3.1 Female Pelvis MRI Dataset

MRI plays a crucial role in the screening and diagnosis of the female pelvis due to its superior soft tissue contrast, high resolution, and absence of ionising radiation. Compared to ultrasound and CT, MRI offers more detailed visualisation of smaller anatomical structures, such as uterine lesions, by accurately depicting their size, location, and morphology. For instance, MRI can detect myomas as small as 0.3 cm in diameter. In the context of uterine myomas, MRI is particularly useful for guiding surgery and determining the nature of the myoma, as some may be suspicious or represent sarcoma (malignant tumour), which can alter surgical planning. For example, identifying a benign myoma allows for less invasive surgical techniques like laparoscopic removal using a morcellator, avoiding large scars or the need for laparotomy, which can be functionally and aesthetically problematic. This is done via various sequences, which emphasise distinct tissue characteristics. In particular, the T2WI sequence is one of the key instruments in identifying pelvic pathologies, providing a clear view of uterine anatomy (Proscia et al., 2010; Sakala et al., 2020; Pan et al., 2024).

Despite these advantages, MRI is not as widely utilised as CT or US in routine clinical settings, primarily due to practical considerations, limited accessibility, and higher costs. Typically, gynaecologists perform a US or CT examination before surgery and request an MRI when myomas are not well visualised or appear suspicious. Although this trend is shifting, there remains a scarcity of publicly available datasets and a lack of extensive research in this area. To bridge this gap and enable learning-based FPMRI segmentation, we have been collecting and annotating a female pelvis MRI segmentation dataset since March 2021. We call this dataset the FPMRId. FPMRId is annotated with the help of both junior and expert radiologists, resulting in 374 segmented medical scans in total, where 201 are already validated and 171 require minor corrections. The information on FPMRId reported in this work corresponds to the period from the beginning of the data collection in March 2021 to October 2024. The process is still ongoing as of October 2024. We first provide an overview of the dataset's properties, followed by a detailed description of the data collection process, including the annotation platforms used, the annotation pipeline, and the annotation guidelines.

Figure 3.1: Evolution of data collection in series from March 2021 to October 2024. 'In-progress' refers to segmented series currently undergoing validation or correction based on feedback.

3.1.1 Overview

Classes. FPMRId includes segmentations of nine classes inside the female pelvis, where six represent anatomical structures: (1) bladder, (2) uterus, (3) uterine cavity, (4) cervix, (5) fundus, and (6) anterior wall. In turn, three remaining classes represent pathologies: (7) uterine myomas, (8) endometriosis, and (9) adenomyosis. The dataset has four key characteristics: (1) it is multi-class, meaning it includes multiple distinct anatomical structures and pathologies, (2) it is multi-label, allowing for overlapping labels where multiple structures or conditions can coexist in the same region (e.g., uterine myomas are located within the uterus), (3) it is multi-instance, meaning several instances of the same class can appear in a single image (e.g., multiple myomas), and (4) it is multi-component, where a single anatomical structure may be represented by multiple disconnected components due to its complex shape and the chosen MRI slice (e.g., multiple non-connected contours of the bladder in a single slice due to to the bladder's shape and slicing).

Annotators. The annotation was performed by a team of three radiologists from CHU de Clermont-Ferrand (university hospital), consisting of two junior radiologists and one senior radiologist. Each annotator worked individually in sequence, with one taking over after the other. The junior radiologists contributed approximately 8.55% and 4.84% of the total annotations, respectively, while the senior radiologist completed the majority with approximately 86.61% of the annotations.

Participants. The dataset consists of anonymized MRI scans collected from patients at CHU de Clermont-Ferrand between August 2001 and March 2024, under an agreement with the latter for anonymized data sharing. All the data is anonymized, so specific patient-related information is not available. Most of the participants included in the dataset presented with one or more pelvic pathologies under study—namely uterine myomas, endometriosis, or adenomyosis.

Image acquisition. Images were acquired using MRI scanners from multiple manufacturers. Specifically, GE Healthcare (GE), Siemens, Philips, and Canon. The most frequently used models were the GE Optima MR450w and Siemens MAGNETOM Avanto, which are followed by the GE Discovery MR750, Siemens MAGNETOM Sonata, and GE SIGNA Artist, which were utilised to a

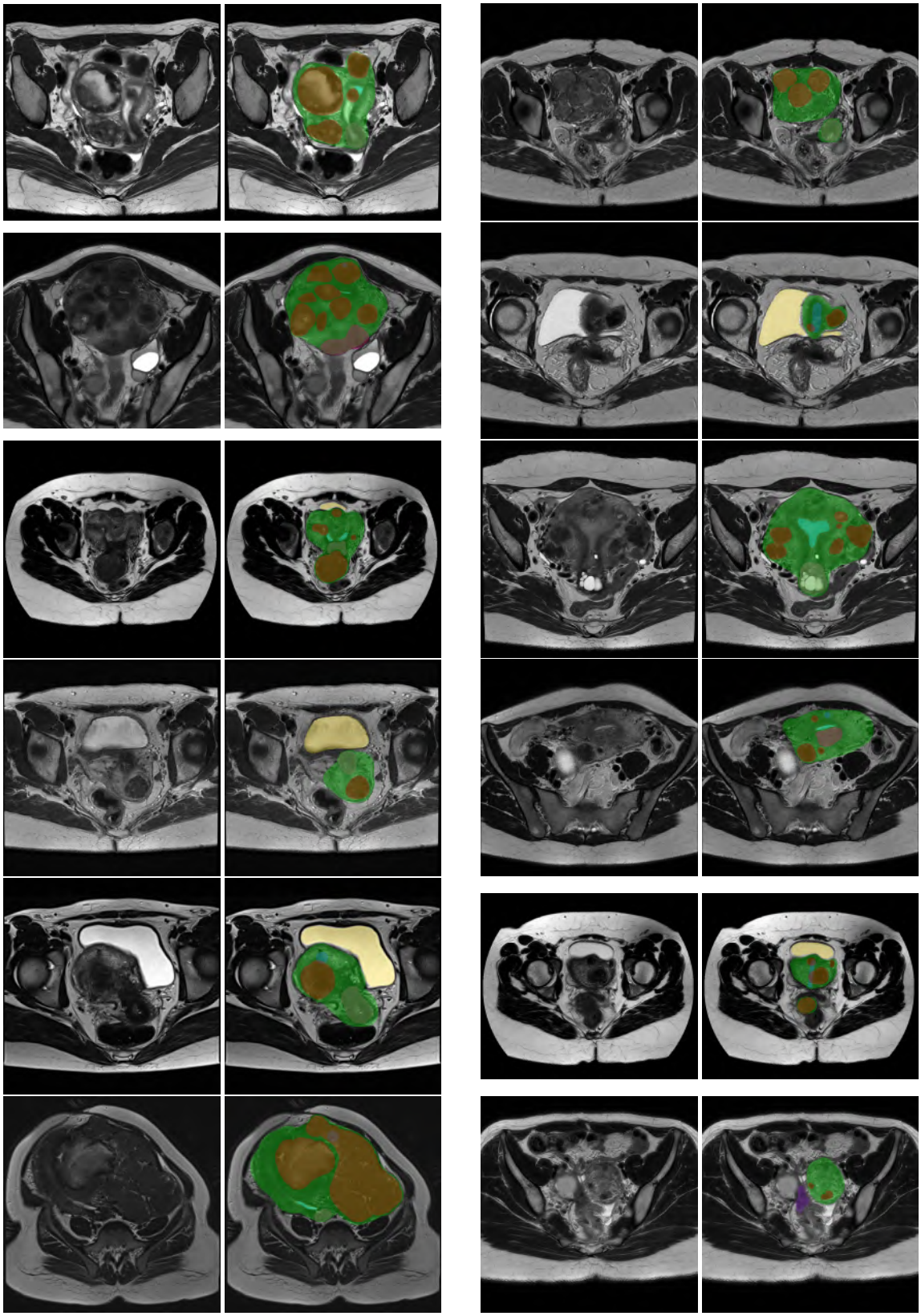


Figure 3.2: 12 MRI slices, with and without segmentations, randomly sampled from FPMRId. Each slice comes from different series. The legend is as follows: (1) bladder - yellow, (2) uterus - light green, (3) uterine cavity - cyan, (4) cervix - pink, (5) fundus - dark green, (6) anterior wall - blue, (7) uterine myomas - red, (8) endometriosis - purple, and (9) adenomyosis - magenta.

Table 3.1: List of MRI machine manufacturers and models included in the FPMRId dataset.

Manufacturer	Model
GE Medical Systems	Optima MR450w
	Discovery MR750
	SIGNA Artist
	SIGNA HDxt
	SIGNA Explorer
	Optima MR360
	SIGNA Voyager
Siemens	MAGNETOM Avanto
	MAGNETOM Sonata
	MAGNETOM Amira
	MAGNETOM Aera
	MAGNETOM Altea
	MAGNETOM Sola
	MAGNETOM Sempra
Philips	Ingenia
	Ingenia Elition X
Canon (Toshiba)	Orian

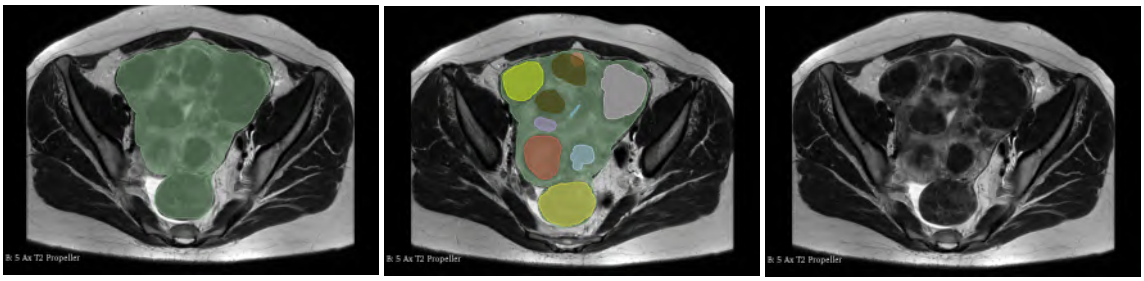
lesser extent. The complete list of models is shown in table 3.1. Scanners operated at magnetic field strengths of either 1.5 Tesla or 3 Tesla. The MRIs were initially taken for clinical indications such as excessive bleeding, infertility, pelvic pain, and cases of endometriosis with myoma for the purpose of myomectomy planning. Myomectomy is a surgical procedure to remove uterine fibroids (non-cancerous growths) while preserving the uterus.

All scans were T2-weighted images, incorporating various specific sequences to optimise image quality and reduce artefacts. The scans were taken in any of the three orientations: sagittal, coronal, or axial. However, axial orientation is dominant in the dataset. Furthermore, some series were acquired in 3D, instead of slice-by-slice 2D acquisition. When done, a three-dimensional imaging sequence, known as CUBE (GE), was employed. Imaging protocols included Turbo Spin Echo (TSE) and Fast Spin Echo (FSE) sequences for rapid acquisition, as well as motion-compensation techniques like BLADE (Siemens) and PROPELLER (GE). Furthermore, Integrated Parallel Acquisition Techniques (IPAT) were employed to accelerate imaging, as well as presaturation techniques were applied in certain cases to suppress unwanted signals.

Imaging parameters are varied across the dataset: slice thickness ranges from 1.2 mm to 6 mm; pixel spacing ranges from 0.2148×0.2148 mm to 1×1 mm; Repetition Time (TR) values range from 1302 ms to 12396 ms, and Echo Time (ET) values range from 75 ms to 143 ms. Image resolution is predominantly 512×512 pixels, followed by 1024×1024 pixels, with a number of scans between 256×256 and 480×480 pixels.

3.1.2 Data Collection

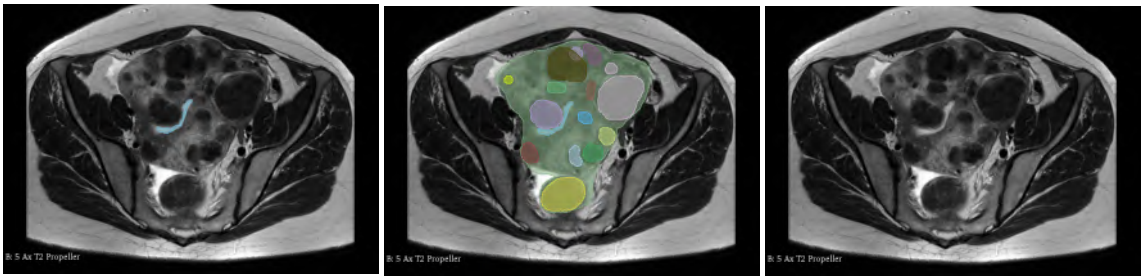
The efficiency of dataset annotation heavily relies on the established annotation pipeline and the annotation platform utilised. An annotation pipeline refers to the sequence of processes involved in annotating the dataset, from patient consent acquisition to the storage of final annotations. The key element of the annotation pipeline is an annotation platform, which is typically a software used to perform the annotation. Simply, the choice of the annotation pipeline setup and specifically of the annotation platform affects annotation throughput and accuracy of the annotations.



Uterus only **All classes** **Raw MRI**

Figure 3.3: An example of Uterus annotation in FPMRId. The same MRI slice is shown three times with only uterus segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in 3D Slicer.

Annotation Platforms. Annotation platforms can be broadly categorised into local, cloud, and hybrid solutions. The choice among these is task-dependent and affects the roles of the actors involved. Local platforms are software suites installed on individual computers, where data is normally stored and processed locally. Cloud platforms are web-based services that store data and annotations on remote servers, enabling remote access and collaboration. Hybrid platforms integrate features from both local and cloud solutions, enabling data processing to occur either locally or in the cloud, depending on the specific functionality being used. For FPMRId, we utilised two local open-source platforms—MITK and 3D Slicer, and one hybrid proprietary platform—Supervisely, throughout the annotation process. Specifically, they respectively account for 13.39%, 71.51% and 15.10% of annotations produced. We cover each of these platforms in turn.



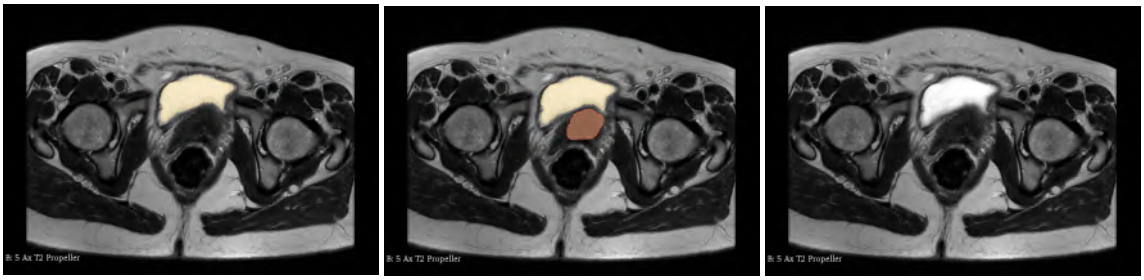
Cavity only **All classes** **Raw MRI**

Figure 3.4: An example of cavity annotation in FPMRId. The same MRI slice is shown three times with only cavity segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in 3D Slicer.

In the initial stages of our project, MITK was employed due to its straightforward GUI and accessibility. However, MITK presented several limitations that affect the annotation workflow. First, it is less actively developed compared to other platforms, leading to technical instability such as frequent crashes and slow performance during the annotation of large series. This instability not only reduces annotation efficiency, but also risks data loss with annotation progress periodically not being saved. Second, MITK lacks collaborative tools. As a result, data management, reviewing the annotations, and providing feedback must be handled through separate solutions, which is inefficient and further increases the risk of errors.

3D Slicer is a more advanced open-source platform. Due to its stability, robust feature set, modular architecture and active development community, 3D Slicer was our primary annotation tool for a significant portion of the project. While 3D Slicer offers significant improvements over

MITK, it still complicates data management and lacks collaborative features. Specifically, both 3D Slicer and MITK operate entirely locally, which means that all data has to be repeatedly manually imported and exported at multiple stages of the annotation pipeline. This process is time-consuming and prone to errors, especially when dealing with large volumes of data. Moreover, 3D Slicer lacks tools for collaborative annotation and version control, making the review, feedback, and correction process particularly cumbersome. Specifically, multiple review rounds require repeated data transfer and external communication, extending the time needed to reach a consensus on the annotations.



Bladder only **All classes** **Raw MRI**

Figure 3.5: An example of bladder annotation in FPMRId. The same MRI slice is shown three times with only bladder segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in 3D Slicer.

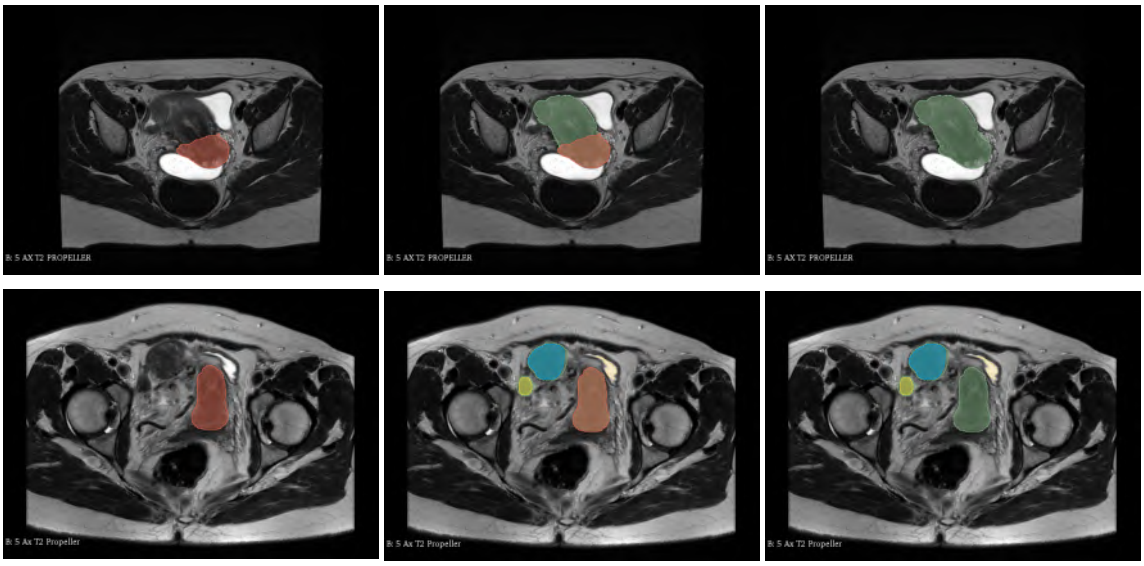
To overcome the limitations of local platforms, we transitioned to Supervisely, a hybrid annotation platform that combines local and online collaborative features. Supervisely allows data to be stored on-premise or in the cloud, while providing an online interface for annotation and team collaboration. This setup enables remote access to the source data and annotation history, from older to the most recent, for both radiologists and research engineers at all times. This eliminates the need to manually import, export, and share data after every change, allowing the annotation process to proceed uninterrupted until completion. Supervisely further enhances the workflow with standardised dataset importing and exporting, data insights, and tools for reviewing and providing feedback, making collaboration much easier. Additionally, the platform's active development team regularly implements user feedback. Specifically, we have provided over 25 feature requests, at least 18 of which were successfully integrated.

Annotation Pipeline. Our annotation pipeline comprises eight steps, where steps from 3 to 7 are platform-dependent. We first present the complete annotation pipeline and then discuss the annotation platform specifics for the steps, where it's applicable. The complete annotation pipeline is as follows:

1. **Medical scan data retrieval and anonymisation:** The radiologist retrieves and anonymises the MRI scans from PACS McKesson (McKesson Corporation, 2024) in DICOM format.

2. **Source data upload to storage:** The radiologist uploads the anonymized MRI scans to a cloud storage platform that holds Health Data Hosting Certification (HDS).

3. **Source data transfer to annotation platform:** The MRI scans are transferred from the storage to the annotation platform. In MITK and 3D Slicer this is done by the radiologist locally, whereas in Supervisely by the research engineer in the cloud.



Cervix only **All classes** **Cervix hidden**

Figure 3.6: Two examples of cervix annotation in FPMRId to showcase how cervix is segmented based on the established guidelines. The same MRI slice is shown three times for each example with only cervix segmentation, all segmentations, and cervix segmentation hidden. Shown segmentations were done in 3D Slicer.

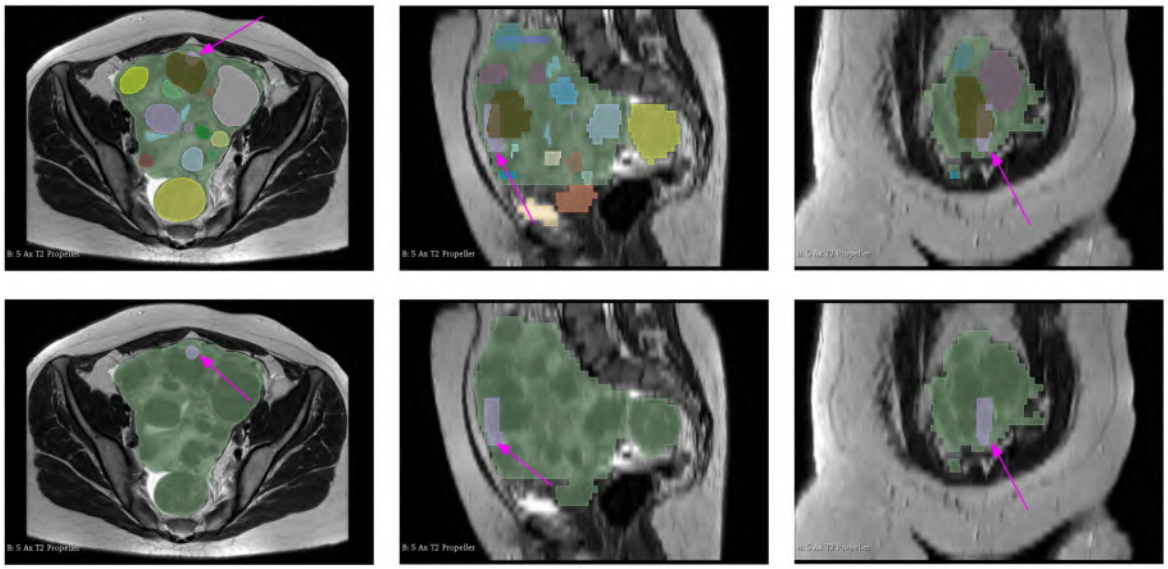
4. **Annotation:** The radiologist segments the MRI scans using the annotation platform.

5. **Annotation transfer to storage** (skipped in Supervisely): In MITK and 3D Slicer the radiologist manually uploads the segmentations produced in step 4 to the storage in step 2. This is skipped in Supervisely, since the segmentations are automatically stored in association with the scans as soon as produced. Simply, in Supervisely the current annotation state is always up to date.

6. **Review, feedback and validation:** The research engineer reviews the segmentations. If the segmentations are done in accordance with the guidelines and meet the required quality, the engineer validates them. The pipeline then proceeds to step 7. However, if further improvements are needed, the engineer provides feedback, and steps 4 to 6 are repeated in an iterative cycle until the consensus on the segmentation is reached. Unlike tools like MITK and 3D Slicer, which do not have built-in support for review, feedback, and validation, Supervisely offers integrated systems specifically designed for these functions.

7. **Complete annotation transfer to storage** (skipped in Supervisely): In MITK and 3D Slicer, the research engineer manually uploads complete annotations to the storage. As with step 5, this is skipped in Supervisely, since the annotations are automatically stored in association with the scans at all times.

The annotation pipeline differs based on whether MITK/3D Slicer or Supervisely is used. In MITK and 3D Slicer, the process follows all steps sequentially. Supervisely, however, optimises the workflow in two key ways. First, data management is streamlined, increasing the efficiency of step 3 and eliminating the need for steps 5 and 7. Second, the review and feedback processes are centralised within the platform, significantly improving the efficiency of step 6. These optimisations are discussed in detail below.



Axial **Sagittal** **Coronal**

Figure 3.7: An example of anterior wall annotation in FPMRId. Two rows are shown with three views (axial, sagittal, and coronal) each for the same MRI slice. Top row shows all segmentations, while the bottom only the uterus and the anterior wall annotations for clarity. The anterior wall segmentation is highlighted by a magenta arrow in all images. Shown segmentations were done in 3D Slicer.

In contrast to local platforms, with Supervisely step 3 is entirely handled by the research engineer, who uploads the source data and creates an annotation job (i.e. a task) for the radiologist using a simple drag-and-drop operation. The source data, resulting annotations, as well as annotation jobs remain in place and consistent at all times. Thus, no further data manipulation is necessary, and steps 5 and 7 are skipped. Compared with MITK and 3D Slicer, this is more efficient for two main reasons: (1) the radiologist does not need to spend time importing, exporting, or sharing data, allowing them to focus more on segmentation, and (2) the job automatically inherits platform settings from the associated annotation project, standardising the process. In our case, this saved approximately 15% of the radiologist's total working time.

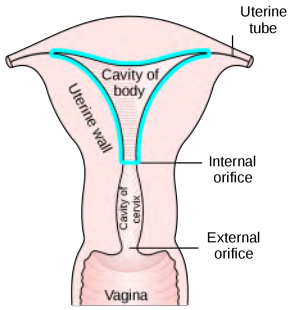


Figure 3.8: A schematic showing the anatomical region annotated as cavity class in FPMRId (in cyan).

For step 6, in the case of MITK and 3D Slicer, the absence of built-in mechanisms necessitates the creation of extensive reports containing screenshots and detailed descriptions of each segmentation issue encountered. The radiologist has to manually correlate these reports with the actual medical scans, identify the problematic areas, and apply the necessary corrections in the annotation platform. This process is extremely time-consuming and introduces potential for errors and omissions due to the manual cross-referencing required. In contrast, Supervisely

significantly enhances the efficiency of reviewing segmentations and providing the feedback by centralising these functions within the platform. Specifically, Supervisely enables the research engineer to mark specific regions of concern directly on the annotations and attach comments. This interactive feedback is immediately accessible to the radiologist, who can navigate to each marked location automatically through the platform's GUI and address the issues one by one without the need to consult external documents. This has resulted in the reduction of the review stage time by 87% on average, depending on the complexity of the medical scan. Overall, switching to Supervisely yielded substantial benefits compared to using MITK and 3D Slicer. Specifically, according to Supervisely's built-in statistics, the total time required to complete the annotation pipeline for a single average medical scan decreased from approximately 80 minutes with MITK and 3D Slicer to about 29.6 minutes with Supervisely, representing a two-thirds reduction in time.

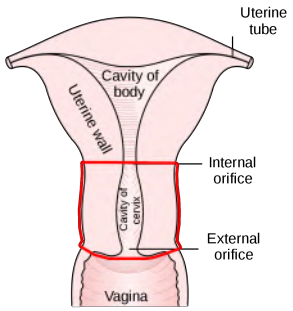
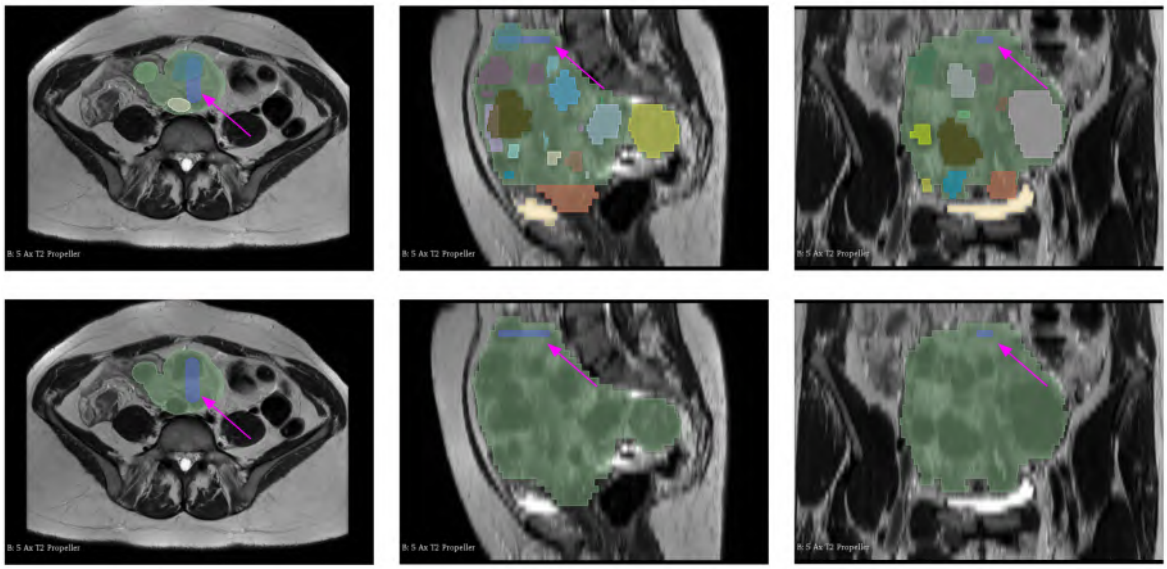


Figure 3.9: A schematic showing the anatomical region annotated as cervix class in FPMRId (in red).

Guidelines. We have established detailed guidelines for the annotation of the nine classes within FPMRId to ensure high precision and consistency across all annotations. We cover these guidelines starting from general guidelines applicable to all classes, followed by specific instruction for each individual class: bladder, uterus, uterine cavity, cervix, fundus, anterior wall, uterine myomas, endometriosis, and adenomyosis.

General. We have established five general rules. First, the radiologist may choose the plane of annotation based on their preference, regardless of whether the MRI scan was acquired in a single plane or through a 3D acquisition. Second, all classes must be fully segmented as precisely as the quality of the data permits across every slice or plane in which they appear. Third, each segmented component must be hole-free, unless anatomically conditioned. Fourth, the radiologist may utilise any tools or methods of their choice within the chosen annotation platform, provided that the second and third rules are respected. Fifth, the segmented regions representing classes inside the uterus must remain entirely within the boundaries of the uterus' segmentation and should not extend beyond them. These classes are: uterine cavity, cervix, fundus, anterior wall, and uterine myomas.

Bladder. For the bladder, we have established two rules. First, the contour of the bladder must be consistently defined across all slices. The bladder typically appears as a lighter, whiter region in T2WI. However, the bladder wall and the surrounding region are often less homogeneous, leading to potential variability in segmentation. Therefore, the choice of the segmentation boundary with respect to the bladder wall should not be inconsistent between slices. Second, the bladder must



Axial **Sagittal** **Coronal**

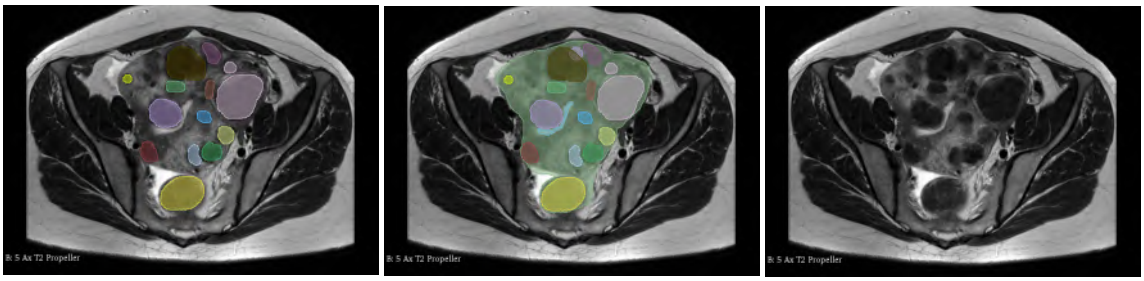
Figure 3.10: An example of fundus annotation in FPMRId. Two rows are shown with three views (axial, sagittal, and coronal) each for the same MRI slice. The top row shows all segmentations, while the bottom row shows only the uterus and the fundus segmentations for clarity. The fundus segmentation is highlighted by a magenta arrow in all images. Shown segmentations were done in 3D Slicer.

not overlap with any other anatomical structures, except in cases where endometriosis invades the bladder. Although endometriosis may extend into the bladder, this has not been observed in our dataset. An example of bladder segmentation is shown in figure 3.5.

Uterus. For the uterus we have established two rules. First, the radiologist must ensure that all internal structures (uterine cavity, cervix, fundus, and anterior wall) are contained within its boundaries. The uterine myomas, endometriosis, and adenomyosis, which may cross the borders of the uterus, must be segmented in accordance with their spread beyond the uterine structure, as these conditions naturally extend beyond the confines of the uterus. Structures like the bladder must not be included within the uterus segmentation. An example of uterus segmentation is shown in figure 3.3.

Uterine cavity. For the uterine cavity, we have established two rules. First, only the cavity of the body of the uterus (corpus) is to be annotated under this label, extending from the fundus down to the internal os, which is the opening between the cervix and the uterine corpus. The cervical canal should not be included into this segmentation, as depicted in figure 3.8. Second, the uterine cavity must not overlap with any other anatomical structures, except for the fundus, anterior wall and adenomyosis. An example of Uterine cavity segmentation is shown in figure 3.4.

Cervix. For the cervix, two rules are established. First, the cervix must be fully segmented within the boundaries of the uterus segmentation, covering the entire region between the internal os and the external os. This region is shown in figure 3.9. Second, the cervix might overlap with other structures, except for the uterine cavity, as outlined in the uterine cavity segmentation guidelines. An example of cervix segmentation is shown in figure 3.6.

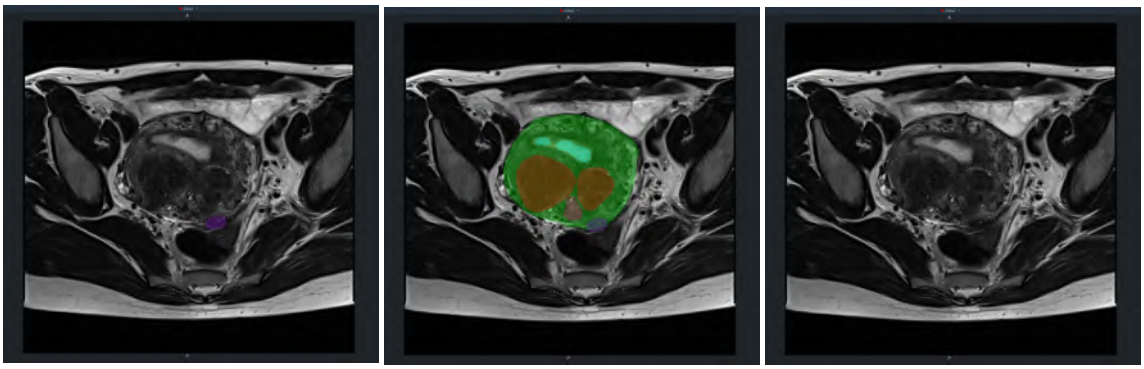


Myomas only **All classes** **Raw MRI**

Figure 3.11: An example of myoma annotation in FPMRId. The same MRI slice is shown three times with only myomas segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in 3D Slicer.

Fundus and Anterior Wall. The fundus and anterior wall are grouped together due to the similarity in their segmentation methodologies, both serving as landmarks rather than exact anatomical representations. Two key rules guide their segmentation. First, both structures are defined by their medial portions, focusing on the central, inner regions within the overall boundaries of the uterus segmentation. Second, the fundus and anterior wall may overlap with other classes within the uterus, as these structures are frequently influenced by pathological changes, such as uterine deformation due to myomas. For instance, in cases with the extensive presence of myomas, the segmentation may overlap with myomas that distort the uterine shape, yet the segmented regions still reflect the appropriate anatomical landmarks, as illustrated in figure 3.10 for fundus and figure 3.7 for anterior wall.

Uterine myomas. For uterine myomas two rules are established. First, all tumours that are anatomically connected to the uterus are segmented as part of the overall uterus segmentation. Second, each tumour must be segmented individually, ensuring there is no overlap with other classes within the uterus, except for the fundus and anterior wall. If multiple tumours are in close proximity and come into contact, they are to be considered as a single tumour and segmented accordingly. An example of uterine myomas segmentation is shown in figure 3.11.

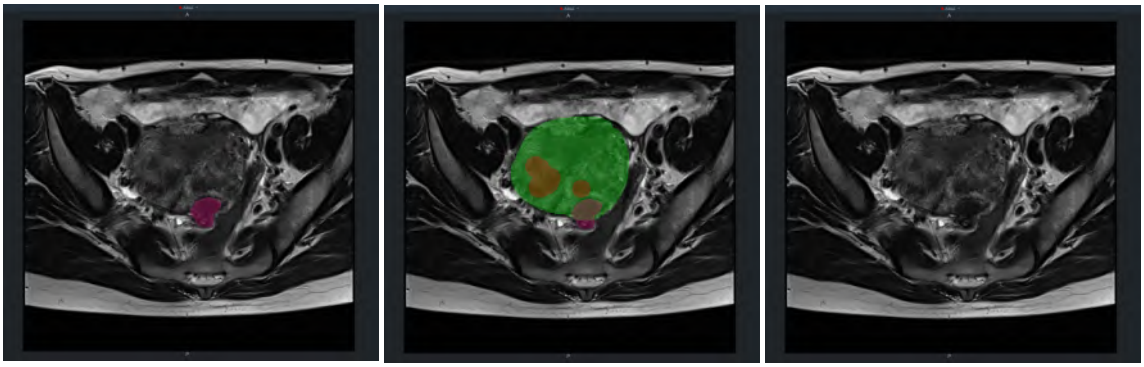


Endometriosis only **All classes** **Raw MRI**

Figure 3.12: An example of endometriosis annotation in FPMRId. The same MRI slice is shown three times with only endometriosis segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in Supervisely, the colouring scheme slightly differs from that of 3D Slicer.

Endometriosis and Adenomyosis. The endometriosis and adenomyosis are grouped together due

to the similarity in their segmentation methodologies. We have established one rule in regard to their segmentation: both endometriosis and adenomyosis may cross the uterus segmentation boundary in both directions and overlap with other classes due to their distinct pathological behaviours. For endometriosis, this is attributed to the condition's ability to invade nearby structures - for example, the ovaries, the fallopian tubes, the bladder and the tissue lining the pelvis. Furthermore, it might spread beyond the pelvic region as well. Adenomyosis, on the other hand, might affect the uterine cavity, the myometrium (middle layer of the uterine wall) and extend beyond the uterus. The examples of segmentation of endometriosis and adenomyosis are shown in figures 3.12 and 3.13 respectively.



Adenomyosis only **All classes** **Raw MRI**

Figure 3.13: An example of adenomyosis annotation in FPMRId. The same MRI slice is shown three times with only adenomyosis segmentation, all segmentations, and no segmentations respectively. Shown segmentations were done in Supervisely, the colouring scheme slightly differs from that of 3D Slicer.

3.1.3 Results

We report the results of data collection for FPMRId in three ways: (1) data collection statistics, (2) analysis of the data using metadata and 15 metrics, and (3) visualisation of select samples.

Data Collection Statistics. We quantified the progression of data collection attributing the scans to one of the two categories: complete and in-progress. Specifically, complete scans are those that passed all the steps in the annotation pipeline, while in-progress scans are those that have passed at least step 4, but require further corrections. Data collection evolution is shown in figure 3.1, which shows the cumulative number of scans over time. The number of scans annotated per annotation platform is shown in figure 3.15.

Data Analysis. To assess the quality and variability of the dataset, we utilised the MRQy (Sadri et al., 2020) - an open-source tool for MRI quality control. Using MRQy, for each scan we extracted 10 key tags from the series metadata and calculated 15 metrics. The list of tags and the list of metrics with descriptions are provided in figure 3.14, courtesy of the original authors. The parallel coordinate chart for each of the 25 values for all FPMRId series is shown in figure 3.17. In turn, the minimum, the maximum, the mean and the standard deviation for applicable tags are reported in table 3.2 and for the metrics in table 3.3. Further, we have embedded all the 25 parameters into a two-dimensional space using UMAP (McInnes et al., 2018), preserving both pairwise and

Type	Number	Metric	Description
Tags	1	MFR	manufacturer name from the tag info
	2	MFS	magnetic field strength from the tag info
	3	VRX	voxel resolution in x plane
	4	VRY	voxel resolution in y plane
	5	VRZ	voxel resolution in z plane
	6	ROWS	rows value of the volume
	7	COLS	columns value of the volume
	8	TR	repetition time value of the volume
	9	TE	echo time value of the volume
	10	NUM	number of slice images in each volume
Measurements	11	MEAN	mean of the foreground intensity values ($\mu_F = \frac{1}{MN}\sum_{(i,j)} F(i,j)$)
	12	RNG	range of the foreground intensity values ($Range = \max(F) - \min(F)$)
	13	VAR	variance of the foreground intensity values ($Variance = \sigma_F^2$)
	14	CV	coefficient of variation percent: standard deviation over the mean of the foreground intensity values ($\%CV = \frac{\sigma_F}{\mu_F} \times 100$)
	15	CPP	contrast per pixel: mean of the foreground intensity values after applying a 3 × 3 2D Laplacian filter ($CPP = \text{mean}(\text{cov2}(F, f_1)), f_1 = \frac{1}{8}\begin{pmatrix}-1 & -1 & -1\\ -1 & 8 & -1\\ -1 & -1 & -1\end{pmatrix}$)
	16	PSNR	peak signal to noise ratio: of the foreground intensity values ($PSNR = 10\log\frac{\max^2(F)}{\text{MSE}(F,f_2)}$, f_2 is a 5 × 5 median filter)
	17	SNR1	signal to noise ratio: foreground standard deviation divided by background standard deviation ($SNR1 = \frac{\sigma_F}{\sigma_B}$)
	18	SNR2	signal to noise ratio: mean of the foreground patch divided by background standard deviation. Foreground patch is a random 5 × 5 square patch of the foreground image ($SNR2 = \frac{\mu_{FP}}{\sigma_B}$)
	19	SNR3	signal to noise ratio: foreground patch standard deviation divided by the centered foreground patch standard deviation ($SNR3 = \frac{\mu_{FP}}{\sigma_{FP-\mu_{FP}}}$)
	20	SNR4	signal to noise ratio: mean of the foreground patch divided by mean of the background patch. Background patch is a random 5 × 5 square patch of the background image ($SNR4 = \frac{\sigma_{FP}}{\sigma_{BP}}$)
	21	CNR	contrast to noise ratio: mean of the foreground and background patches difference divided by background patch standard deviation ($CNR = \frac{\mu_{FP-BP}}{\sigma_{BP}}$)
	22	CVP	coefficient of variation of the foreground patch: foreground patch standard deviation divided by foreground patch mean ($CVP = \frac{\sigma_{FP}}{\mu_{FP}}$)
	23	CJV	coefficient of joint variation between the foreground and background ($CJV = \frac{\sigma_F+\sigma_B}{\lvert\mu_F-\mu_B\rvert}$)
	24	EFC	entropy focus criterion ($EFC = \frac{NM}{\sqrt{NM}}\log\frac{E}{\sqrt{NM}}, E = -\sum_{(i,j)}\frac{F(i,j)}{F_{max}}\ln\frac{F(i,j)}{F_{max}}, F_{max} = \sqrt{\sum_{(i,j)}F^2(i,j)}$)
	25	FBER	foreground-background energy ratio ($FBER = \frac{\text{median}(\lvert F\rvert^2)}{\text{median}(\lvert B\rvert^2)}$)

All the computed measurements (number 11-25) are average values over the entire volume, which calculated for every single slice separately. Variables M, N, F, B, FP, BP stand for slice width size, slice height size, foreground image, background image, foreground patch, and background patch respectively. Operator μ, σ, σ^2, median stand for mean, standard deviation, and variance measures respectively.

Figure 3.14: Definitions of the 25 parameters used in MRQy, comprising 10 tags extracted from series metadata and 15 metrics calculated for MRI analysis. Source: (Sadri et al., 2020)

Volume status	MITK	3D Slicer	Supervisely	TOTAL
Fully complete	47	97	57	201
In-progress	0	154	19	173
TOTAL	47	251	76	**374**

Figure 3.15: The number of scans annotated using each of the three annotation platforms: MITK, 3D Slicer and Supervisely.

Figure 3.16: A scatter plot visualizing the 25 parameters reported for each series, embedded into a two-dimensional space using Uniform Manifold Approximation and Projection (UMAP). Each dot represents a series from FPMRId.

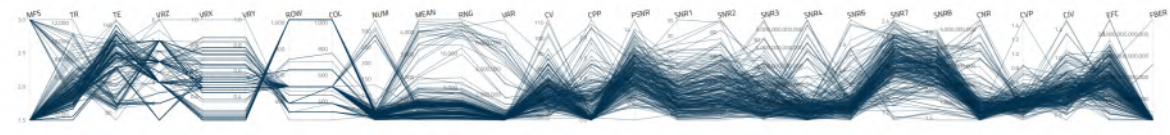


Figure 3.17: The parallel coordinate chart for the 25 MRQy parameters for the entirety of the FPMRId.

global distances between scans. The resulting scatter plot is shown in figure 3.16. We observe several well-separated clusters, indicating the presence of distinct patterns within the data. These patterns are likely influenced by variations in manufacturer, scanner model, and acquisition parameters, resulting in a heterogeneous and diverse dataset.

Table 3.2: The mean, standard deviation, minimum and maximum for the 8 numerical tags extracted from the metadata across all series in FPMRId.

	TR	TE	VRZ	VRX	VRY	ROW	COL	NUM
MEAN	4651.16	115.92	3.52	0.49	0.49	572.60	575.64	49.48
STDDEV	1930.83	16.56	1.01	0.14	0.14	197.11	195.56	52.14
MIN	1302.00	75.00	1.20	0.2148	0.2148	256.00	256.00	23.00
MAX	12396.78	143.81	6.00	1.00	1.00	1024.00	1024.00	336.00

Table 3.3: The mean, standard deviation, minimum and maximum for the 25 metrics calculated across all series in FPMRId.

	MEAN	RNG	VAR	CV	CPP	PSNR	SNR1	SNR2	SNR3	SNR4	SNR6	SNR7	SNR8	CNR	CVP	CJV	EFC	FBER
MEAN	860.20	2931.75	5.66×10^{5}	66.00	0.35	10.41	11.10	33.39	10.38	9.00×10^{11}	1.40	1.85	2.35	9.00×10^{8}	0.33	0.77	2.74	8.87×10^{10}
STDDEV	896.30	3193.04	1.33×10^{6}	11.40	0.49	1.62	5.36	15.73	6.14	1.02×10^{12}	0.34	0.31	0.34	1.02×10^{9}	0.14	0.13	0.37	7.29×10^{11}
MIN	75.84	298.23	3.27×10^{3}	47.69	0.00	6.60	3.22	8.99	2.51	3.11×10^{10}	1.03	1.10	1.47	3.11×10^{7}	0.07	0.56	2.07	2.90×10^{1}
MAX	4564.47	14924.10	7.83×10^{6}	111.96	2.43	14.83	35.29	90.30	36.94	5.54×10^{12}	2.99	2.42	3.22	5.54×10^{9}	1.48	1.27	3.74	9.38×10^{12}

Select Sample Visualisation. We randomly sample one segmented image from each of twelve scans. These images are displayed in figure 3.2 in pairs: with and without their corresponding segmentations. In this work FPMRId is utilised in three ways: (1) a large number of experiments involving model training and evaluation, (2) user evaluation study, and (3) inter-expert variability study. First, the dataset served as the foundation for training and validating segmentation models developed for interactive neural segmentation and concurrent data-efficient annotation and model training, which constitute the two key contributions of this thesis in sections 4 and 5 respectively. Second, we conducted a user evaluation involving eight medical experts who interactively used the segmentation model trained on the FPMRId dataset, as reported in section 4. Third, we performed an inter-expert variability study to assess the segmentation consistency between medical professionals with different expertise levels. This study is presented in detail in the next section.

3.2 Inter-Expert Variability Study

Medical image annotation is challenging due to the inherent complexities of medical images, as well as human factors. Specifically, while medical scans may exhibit anatomical ambiguities, noise and artefacts, the differences of experience, attentiveness and fatigue of the medical expert play a big role in the final quality of the annotations. Consequently, annotations performed by different medical experts may vary significantly for the same scan. This variability among experts is referred to as inter-expert or inter-observer variability in the literature. Available studies mostly investigate inter-expert variability for more common targets and imaging modalities, particularly in CT scans. For instance, inter-observer variability has been studied in the segmentation of brain tumours (Jungo et al., 2018), prostate (Montagne et al., 2021), bladder cancer (Foroudi et al., 2009),

and other structures in CT imaging (Woo et al., 2020; Joskowicz et al., 2019). CT remains the dominant modality in such studies, with MRI less frequently addressed. Moreover, there is a paucity of studies focusing on inter-observer variability in the segmentation of the female pelvis in MRI, which is a complex domain due to MRIs exhibiting a high level of variation. Some work has been done on the segmentation of pelvic bones in CT (Juergensen et al., 2024) and defining uterine position using ultrasound imaging (Baker et al., 2013). However, we have not found inter-observer variability studies specifically targeting MRI segmentation of the uterus and its internal anatomical structures. This is significant because MRI, particularly T2WI, is one of the key imaging techniques for diagnostics of female pelvis pathologies.

In view of the above, we conducted a retrospective, single-centre inter-expert variability study for the segmentation of female pelvis MRI using the data from FPMRId. The primary objective of this study is to analyse the segmentation correlation among radiologists with varying levels of experience to determine the reliability and reproducibility of this process for 5 segmentation classes of female pelvis MRI. Secondary objectives include assessing experts' diagnostic performance and investigating potential links between segmentation precision and the expert's experience. One of the strengths of this study is the number of participating experts, which is six, as opposed to three, which is common in the literature (Dissaux et al., 2020; Lim et al., 2011; Rosa et al., 2020). We present the methodology and the obtained results in the following sections.

MEAN

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		0.930	0.930	0.911	0.924	0.927
RR2	0.930		0.921	0.901	0.914	0.922
RJ1	0.930	0.921		0.913	0.926	0.923
RJ2	0.911	0.901	0.913		0.920	0.904
RS1	0.924	0.914	0.926	0.920		0.920
RS2	0.927	0.922	0.923	0.904	0.920	

STDDEV

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		0.021	0.028	0.043	0.030	0.027
RR2	0.021		0.025	0.046	0.033	0.023
RJ1	0.028	0.025		0.034	0.019	0.018
RJ2	0.043	0.046	0.034		0.025	0.035
RS1	0.030	0.033	0.019	0.025		0.020
RS2	0.027	0.023	0.018	0.035	0.020	

CV

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		0.023	0.030	0.047	0.033	0.029
RR2	0.023		0.028	0.052	0.036	0.025
RJ1	0.030	0.028		0.037	0.020	0.020
RJ2	0.047	0.052	0.037		0.028	0.039
RS1	0.033	0.036	0.020	0.028		0.022
RS2	0.029	0.025	0.020	0.039	0.022	

Figure 3.18: Pairwise dice values for uterus across all 10 series: MEAN, standard deviation as STDDEV, and Coefficient of Variation (CV). Grey zone contains mirrored values.

3.2.1 Methodology

Data Collection. We use a subset of 10 MRI series, randomly sampled from the FPMRId dataset. These 10 series were collected from adult female patients, who underwent pelvic MRI as part of a pre-surgical evaluation prior to myomectomy. The MRI scanner used is GE SIGNA ARTIST with the magnetic field strength of 1.5 Tesla. The imaging protocol is T2WI PROPELLER with a slice thickness of 5 mm and an image resolution of 512×512 pixels.

Data Annotation. Each of the 10 MRI series is segmented independently by six radiologists with varying levels of experience. Based on their expertise, the radiologists are divided into three groups: (1) two radiology interns, (2) two assistant radiologists specialising in women's imaging, and (3) two senior hospital practitioners with three and seven years of experience, respectively, who specialise in the female pelvis. For clarity, each expert is referred to as Junior Radiologist (RJ), Radiology Resident (RR), or Senior Radiologist (RS) based on their group respectively. The experts are thus labelled as RR1 and RR2, RJ1 and RJ2 and RS1 and RS2. All the radiologists were tasked with segmenting five classes of the female pelvis: (1) uterus, (2) bladder, (3) cervix, (4) uterine cavity and (5) uterine myomas. All segmentation tasks were performed manually or semi-automatically using 3D Slicer with no limitations on the tools used. Notably, no DL models were employed. To avoid the influence of prior experience with 3D Slicer on final segmentation results, all radiologists underwent standardised training on the use of 3D Slicer. To ensure proficiency, the very first MRI segmentation for each expert was supervised by an expert already proficient with 3D Slicer.

Data Analysis. Since this study features six experts, including two senior radiologists, each expert's unique experience and interpretation may result in differing, but plausible, segmentations. Simply, no single segmentation can be considered the golden standard in this study by definition. However, to effectively assess the performance of each expert, a consensus-based reference segmentation is crucial. Therefore, for the analysis we adopt a dual approach: (1) we perform pairwise comparisons between expert segmentations to directly assess inter-expert agreement and capture relative variability and consistency, and (2) we generate a reference segmentation that synthesises the contributions of all experts to serve as a benchmark for evaluation. Specifically, this study is divided into three key steps: (1) calculation of segmentation metrics, (2) generation of a reference segmentation, and (3) statistical analysis of the metrics from step 1 in relation to the reference segmentation from step 2. We review each of these steps in turn.

Segmentation Metrics. Two primary metrics are calculated: the Dice Similarity Coefficient (Dice) and the volume (in cm^3). The metrics are chosen to be complementary. Specifically, the dice estimates the degree of similarity between two segmentations directly, but does not account for the actual size of the anatomical structures. In contrast, volume provides an indication of the anatomical structure's size. These metrics are calculated to assess the agreement between experts for each class and series, as well as across all classes and series using the mean, standard deviation and CV. For these calculations, all classes, with the exception of uterine myomas, were treated uniformly. This distinction is necessary because an expert can identify an arbitrary number of myomas in a single series, which may not match the number of myomas identified by another expert. To address this, in addition to the standard dice calculation, we propose and calculate the Found Myoma Agreement (FMA) metric for each series to estimate the degree of agreement on the number of myomas present in a series among the experts. Both dice and volume metrics were calculated using the "Segment Comparison" module of 3D Slicer as a part of the SlicerRT toolkit (Pinter et al., 2012). In turn, the FMA, the mean, the standard deviation and the CV, were calculated using Google Sheets. Overall, the study resulted in 46 sheets, containing over 400 tables. The methodology for calculating dice, volume and FMA is discussed below.

MEAN

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		11.64	14.12	25.52	22.10	18.42
RR2	11.64		16.94	27.49	26.62	21.97
RJ1	14.12	16.94		25.55	19.17	28.57
RJ2	25.52	27.49	25.55		17.60	38.19
RS1	22.10	26.62	19.17	17.60		35.00
RS2	18.42	21.97	28.57	38.19	35.00	

STDDEV

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		7.81	11.79	27.01	18.17	27.09
RR2	7.81		13.73	27.47	23.83	32.11
RJ1	11.79	13.73		30.14	23.76	38.43
RJ2	27.01	27.47	30.14		15.52	43.55
RS1	18.17	23.83	23.76	15.52		36.44
RS2	27.09	32.11	38.43	43.55	36.44	

CV

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
RR1		0.67	0.83	1.06	0.82	1.47
RR2	0.67		0.81	1.00	0.90	1.46
RJ1	0.83	0.81		1.18	1.24	1.35
RJ2	1.06	1.00	1.18		0.88	1.14
RS1	0.82	0.90	1.24	0.88		1.04
RS2	1.47	1.46	1.35	1.14	1.04	

Figure 3.19: Pairwise absolute volume differences for uterus across all 10 series: MEAN, standard deviation as STDDEV, and CV. Grey zone contains mirrored values.

The dice measures the overlap between two sets of data, ranging from 0 (no overlap) to 1 (perfect overlap). It is calculated according to the equation 3.1.

$$\text{Dice}(A,B) = \frac{2 \cdot |A \cap B|}{|A| + |B|} \tag{3.1}$$

We calculate the dice for each class using the following two-step procedure: (1) for each series, 15 pairwise dice values are calculated, since 15 unique pairs can be formed from 6 experts, calculated as $\frac{6\times5}{2} = 15$), (2) the mean, the standard deviation and the CV are calculated across all series, yielding 15 mean, standard deviation and CV values respectively. For uterine myomas, this procedure is applied only to myomas that were segmented by at least two experts (referred to as "agreed-upon myomas"). A table displaying the uterus dice values for all 10 series, along with their respective means, standard deviations and coefficients of variation, is presented in figure 3.18.

To compare the volumes of segmentations, we use the following three-step procedure: (1) for each series, six volume values are calculated, corresponding to the number of experts, (2) 15 pairwise absolute volume differences are calculated, which means a single value for each expert pair, (3) the mean, the standard deviation and the CV are calculated across all series, yielding 15 mean, standard deviation and CV values respectively. A table with the uterus volume values for all 10 series can be seen in figure 3.20. In turn, means, standard deviations and coefficients of variation for absolute volume differences for these 10 series can be seen in figure 3.19.

Due to the variability in the number of myomas identified by each expert, directly calculating dice and volume may not accurately reflect expert performance. For instance, the discrepancy between two experts could increase if one expert identified more myomas than the other, which is frequently observed as shown in figure 3.21. Therefore, for uterine myomas the metrics are calculated as follows: (1) a medical expert visually examines and matches the myomas segmented by each expert, assigning the same identifier to each distinct myoma across all experts, (2) the dice and volume metrics are calculated only for agreed-on myomas and indicate the segmenta-

Expert	RR1	RR2	RJ1	RJ2	RS1	RS2	MEAN	STDDEV	CV
series1	81.82	77.98	83.82	99.75	82.12	82.31	84.63	7.66	0.09
series2	337.44	322.75	352.60	345.01	341.33	310.84	335.00	15.43	0.05
series3	129.73	123.28	129.19	131.24	141.76	124.03	129.87	6.65	0.05
series4	774.43	776.47	762.99	862.87	828.99	752.79	793.09	43.16	0.05
series5	829.46	842.94	867.87	857.15	858.14	738.51	832.34	47.89	0.06
series6	262.64	275.24	280.25	289.89	285.21	264.13	276.23	11.09	0.04
series7	150.02	142.50	159.76	205.43	172.90	149.04	163.27	23.21	0.14
series8	504.46	486.63	533.28	492.24	523.63	493.90	505.69	18.80	0.04
series9	933.15	904.37	921.51	942.31	982.53	951.86	939.29	26.87	0.03
series10	154.26	163.48	160.12	162.29	161.76	161.30	160.53	3.27	0.02

Figure 3.20: Uterus volumes for all 10 series, as obtained by each of the 6 experts. Also shown: MEAN, standard deviation as STDDEV, and CV. CV is a fraction.

tion consistency level between the experts, while (3) the FMA metric is calculated to estimate the expert's consensus on myoma number. When used jointly, these metrics allow a more complete view on the expert's performance. For example, a high dice score, but low FMA score may indicate that agreed-upon myomas were segmented very similarly by the experts, but the experts strongly disagree on the number of myomas in the series. FMA is calculated for each series using the following three-step process: (1) the number of myomas jointly found by all the experts is considered to be the maximum number of myomas in a series, (2) each time an expert identifies a myoma or misses a myoma, the respective count is increased, resulting in two values per series, and (3) using the values from step 2, the FMA is calculated according to the formula 3.2. Specifically:

$$\text{FMA} = \frac{F}{M+F}, \tag{3.2}$$

where F and M are the numbers of experts who found or missed a myoma respectively. This results in a single FMA score per series, which represents the agreement rate between the experts. An FMA table for a single series with 10 myomas can be seen in figure 3.22.

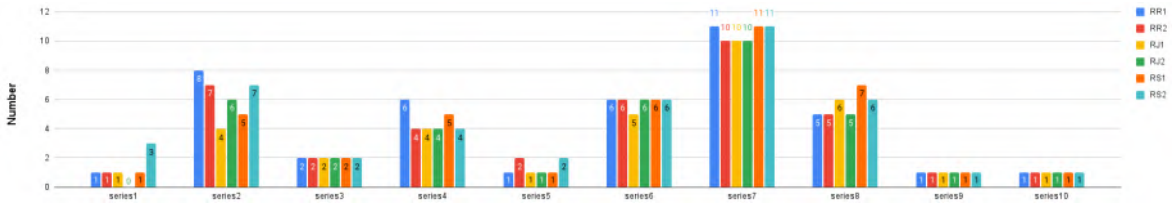


Figure 3.21: The number of myomas identified by each expert in each of the 10 series.

Reference Segmentation. We use STAPLE (Warfield et al., 2004), an iterative weighted voting algorithm, to generate a reference golden standard-like segmentation by merging the segmentations of all the experts for each series. Specifically, STAPLE addresses the challenge of variability among individual segmentations by iteratively estimating both the consensus segmentation and the performance level of each expert. STAPLE operates in four general steps: (1) all segmentations are merged by majority vote to create a preliminary estimate of the consensus segmentation (2) the accuracy of each expert's segmentation is evaluated against this estimate, assigning weights based on expert's performance, (3) the consensus segmentation is updated by weighting each expert's input accordingly and (4) steps 2-3 are repeated until convergence. While STAPLE provides a stable way to merge variable input from multiple medical expects, it has three main drawbacks: (1) it may dismiss the input of an objectively correct expert if their segmentation differs from the

Myoma	RR1	RR2	RJ1	RJ2	RS1	RS2	Found	Missed	FMA	FMA TOTAL
M1	1	1	1	1	1	1	6	0	1.00	
M2	1	1	1	1	1	1	6	0	1.00	
M3	1	1	0	1	1	1	5	1	0.83	
M4	1	1	1	1	1	1	6	0	1.00	
M5	1	1	1	0	0	1	4	2	0.67	0.62
M6	1	1	0	0	0	1	3	3	0.50	
M7	0	1	0	1	0	1	3	3	0.50	
M8	0	0	0	1	1	0	2	4	0.33	
M9	1	0	0	0	0	0	1	5	0.17	
M10	1	0	0	0	0	0	1	5	0.17	

Figure 3.22: FMA scores for series 2, based on whether each expert identified or missed specific myomas. Found and missed myomas, as well as their quantities are indicated.

majority, potentially overlooking accurate results simply because they are in the minority, (2) it tends to underestimate organ boundaries because the edges often have lower consensus among raters, leading the algorithm to exclude precise contours provided by some of the experts, and (3) the algorithm operates solely based on consensus without considering image properties or anatomical continuity, which can result in segmentations that have discontinuities or holes in structures that should be continuous. Figure 3.23 presents examples of consensus segmentations generated with STAPLE for 8 slices from different series, along with a reference segmentation by RS2 and a heatmap overlay of all experts' segmentations.

Statistical Analysis. Three methods are used to infer the relationship between the experts' segmentations and STAPLE consensus segmentations. These methods are Spearman correlation (Spearman, 1904), Kruskal-Wallis test (Kruskal and Wallis, 1952) and post-hoc Dunn-Bonferroni test (Dunn, 1961), which is applied depending on the Kruskal-Wallis test result. Each of these tests were conducted using the open-source software JASP (JASP Team, 2024). We describe each in turn.

First, we begin by measuring Spearman correlation. This test aims to determine whether a correlation exists, for each class, between the expert's segmentation similarity to the STAPLE consensus segmentation and the expert's level of experience. In this context, the Spearman test has two inputs: (1) the set of 10 Dice scores (1 per series) for each expert, representing the similarity between the expert's segmentation and the STAPLE consensus as shown in figure 3.26, and (2) the experience level score of each expert. For the latter, each of the experts is assigned an experience level score from 1 to 6 as follows: RR1 = 1, RR2 = 2, RJ1 = 3, RJ2 = 4, RS1 = 5 and RS2 = 6. Spearman test outputs two scores: r_s - the correlation coefficient, and the p-value. r_s measures the strength and direction of a relationship between two variables. It ranges from -1 to 1, where values close to 1 or -1 indicate strong positive or negative correlations, respectively, while values near zero suggest little to no correlation. In turn, p-value assesses the statistical significance of the observed correlation. Specifically, p-value below a predetermined significance threshold, indicates that the observed correlation is unlikely to be due to random chance, suggesting a statistically significant relationship. Conversely, a high p-value implies that the correlation is not statistically significant. We set the significance threshold to 0.05, following standard practice.

Second, we perform the Kruskal-Wallis test (Kruskal and Wallis, 1952). This test aims to determine whether statistically significant differences exist among the six experts with respect to the

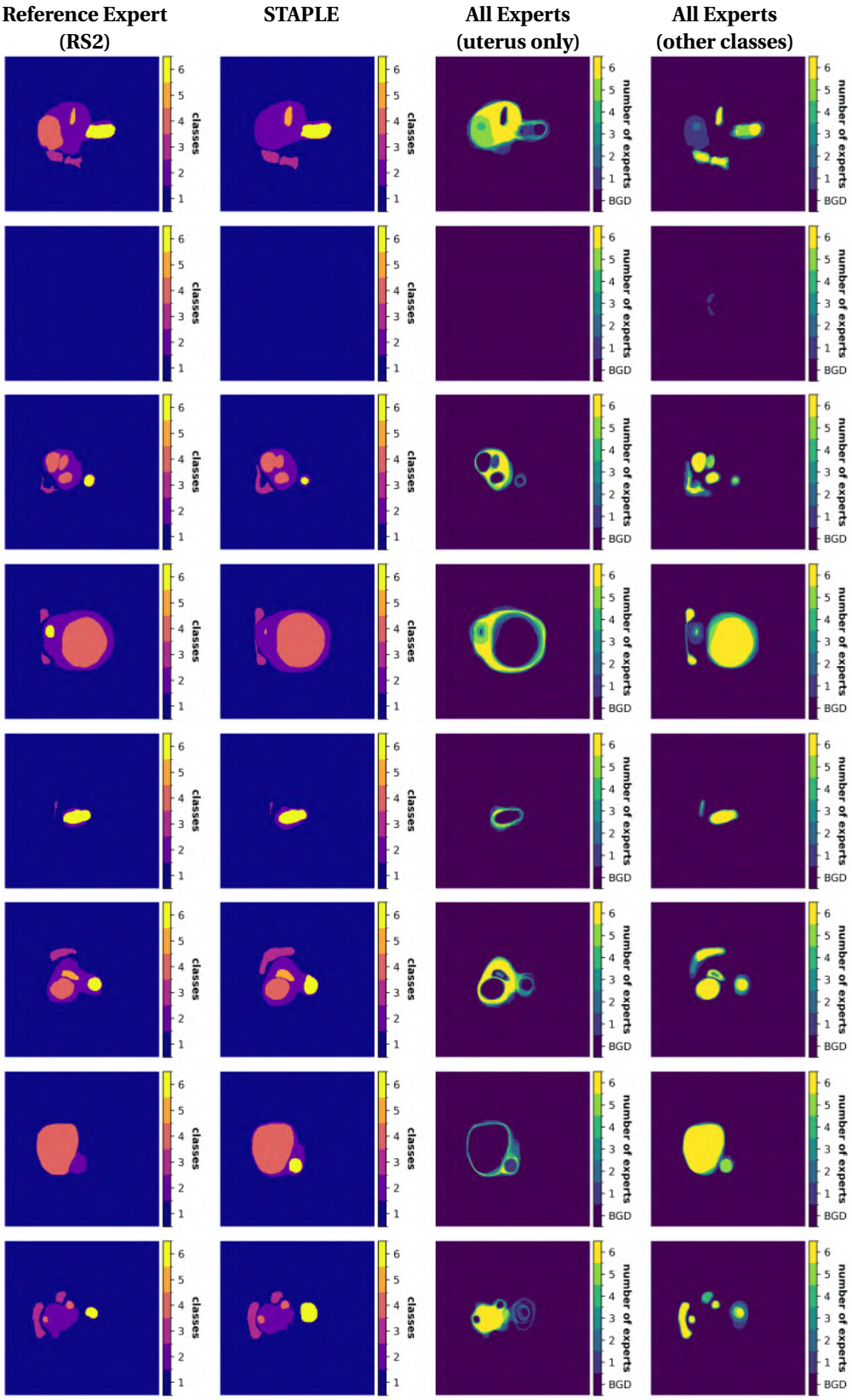


Figure 3.23: Consensus segmentations generated with STAPLE for 8 slices from different series, with supporting information. The columns are: (1) reference segmentation by RS2, (2) STAPLE consensus segmentation, (3) heatmap of experts' segmentations for the uterus only, and (4) heatmap of experts' segmentations for other classes.

Series	Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
series1	RR1		0.933	0.918	0.882	0.929	0.930
	RR2	0.933		0.909	0.868	0.923	0.923
	RJ1	0.918	0.909		0.892	0.929	0.913
	RJ2	0.882	0.868	0.892		0.890	0.890
	RS1	0.929	0.923	0.929	0.890		0.926
	RS2	0.930	0.923	0.913	0.890	0.926	
series2	RR1		0.927	0.915	0.922	0.937	0.921
	RR2	0.927		0.905	0.906	0.921	0.930
	RJ1	0.915	0.905		0.913	0.916	0.898
	RJ2	0.922	0.906	0.913		0.920	0.914
	RS1	0.937	0.921	0.916	0.920		0.919
	RS2	0.921	0.930	0.898	0.914	0.919	
series3	RR1		0.923	0.940	0.922	0.915	0.924
	RR2	0.923		0.921	0.898	0.895	0.917
	RJ1	0.940	0.921		0.918	0.921	0.922
	RJ2	0.922	0.898	0.918		0.917	0.908
	RS1	0.915	0.895	0.921	0.917		0.901
	RS2	0.924	0.917	0.922	0.908	0.901	
series4	RR1		0.951	0.949	0.919	0.937	0.955
	RR2	0.951		0.942	0.918	0.933	0.949
	RJ1	0.949	0.942		0.913	0.922	0.945
	RJ2	0.919	0.918	0.913		0.924	0.913
	RS1	0.937	0.933	0.922	0.924		0.933
	RS2	0.955	0.949	0.945	0.913	0.933	
series5	RR1		0.949	0.949	0.943	0.950	0.930
	RR2	0.949		0.940	0.935	0.942	0.920
	RJ1	0.949	0.940		0.942	0.949	0.911
	RJ2	0.943	0.935	0.942		0.947	0.913
	RS1	0.950	0.942	0.949	0.947		0.911
	RS2	0.930	0.920	0.911	0.913	0.911	

Figure 3.24: Pairwise dice values for uterus for 5 series. Grey zones contain mirrored values.

STAPLE consensus segmentation. In this context, this test inputs and the output of the Kruskal-Wallis test repeat in part those of Spearman correlation. Specifically, this test inputs 15 dice or volume difference scores, as shown in figure 3.24 for dice and figure 3.25 for volume. Each score represents the mean across all series between a pair of experts. The Kruskal-Wallis test outputs a p-value, indicating whether there are significant differences between experts. If the p-value is below the significance threshold (0.05), it suggests that at least two experts differ significantly. However, identifying which specific experts differ requires a post-hoc test. Third, if the Kruskal-Wallis test reveals significant differences between experts, we perform a post-hoc Dunn-Bonferroni test (Dunn, 1961) to identify which specific pairs of experts differ. The Dunn-Bonferroni test takes as input the same set of dice or volume difference scores used in the Kruskal-Wallis test and outputs p-values for each pair of experts, indicating which pairs show statistically significant differences. The same significance threshold (0.05) is used. While this test pinpoints the expert pairs with differing segmentations for the same class, it does not explain the reasons for these differences, which are up to interpretation.

3.2.2 Results

We present and discuss the results of the inter-expert variability study for each class in the following order: (1) uterus, (2) bladder, (3) cervix, (4) uterine cavity, and (5) uterine myomas. For each class, we provide segmentation metrics and statistical analysis results. The segmentation metrics are reported in five formats: (1) a table showing the mean dice scores for each expert in comparison with other experts, (2) a scatter plot displaying dice scores for each expert/series compared to the STAPLE consensus segmentation, (3) a table with mean dice scores based on the scatter plot in (2), (4) a bar chart illustrating volume measurements for each expert/series, and (5) a table showing the mean volumes for each series across experts, derived from the bar chart in (4). For uterine myomas, we provide two additional bar charts: (1) the number of myomas identified for each expert/series, and (2) an FMA score for each series. In turn, the statistical results are compiled into a single table for all classes, which includes the outcomes of Spearman correlation, Kruskal-Wallis test, and Dunn-Bonferroni post-hoc test, where applicable.

Series	Expert	RR1	RR2	RJ1	RJ2	RS1	RS2
series1	RR1		3.84	2.00	17.93	0.30	0.48
	RR2	3.84		5.84	21.77	4.14	4.32
	RJ1	2.00	5.84		15.93	1.70	1.52
	RJ2	17.93	21.77	15.93		17.63	17.45
	RS1	0.30	4.14	1.70	17.63		0.18
	RS2	0.48	4.32	1.52	17.45	0.18	
series2	RR1		14.69	15.17	7.57	3.89	26.60
	RR2	14.69		29.85	22.26	18.58	11.91
	RJ1	15.17	29.85		7.59	11.27	41.76
	RJ2	7.57	22.26	7.59		3.68	34.17
	RS1	3.89	18.58	11.27	3.68		30.49
	RS2	26.60	11.91	41.76	34.17	30.49	
series3	RR1		6.44	0.54	1.51	12.03	5.70
	RR2	6.44		5.91	7.96	18.47	0.75
	RJ1	0.54	5.91		2.05	12.57	5.16
	RJ2	1.51	7.96	2.05		10.52	7.21
	RS1	12.03	18.47	12.57	10.52		17.73
	RS2	5.70	0.75	5.16	7.21	17.73	
series4	RR1		2.04	11.44	88.44	54.56	21.64
	RR2	2.04		13.48	86.40	52.52	23.68
	RJ1	11.44	13.48		99.88	66.00	10.20
	RJ2	88.44	86.40	99.88		33.88	110.08
	RS1	54.56	52.52	66.00	33.88		76.20
	RS2	21.64	23.68	10.20	110.08	76.20	
series5	RR1		13.48	38.41	27.69	28.68	90.95
	RR2	13.48		24.93	14.21	15.20	104.43
	RJ1	38.41	24.93		10.72	9.73	129.36
	RJ2	27.69	14.21	10.72		0.99	118.64
	RS1	28.68	15.20	9.73	0.99		119.63
	RS2	90.95	104.43	129.36	118.64	119.63	

Figure 3.25: Pairwise absolute volume differences for uterus for 5 series. Grey zones contain mirrored values.

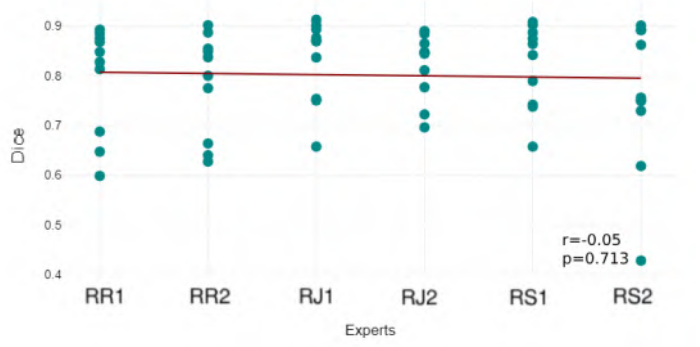


Figure 3.26: Expert's segmentation vs. the STAPLE consensus for uterus segmentation for each of the 10 series: 10 Dice scores (1 per series) for each expert. The red line represents the simple linear regression model.

Uterus

Segmentation Metrics. The assessment of segmentation consistency of the uterus revealed an excellent inter-expert correlation, irrespective of the expert involved. The mean dice score across all expert comparisons is 0.919 ± 0.029 as reported in table 3.4. The low standard deviations and coefficients of variation further confirm the minimal variation in segmentation performance between experts. When comparing expert segmentations against the STAPLE consensus segmentation, the mean dice score is observed to be lower at 0.801 ± 0.101 as reported in table 3.5. Despite this reduction, the results still indicate a strong alignment with the consensus, supported by the detailed scatter plot of dice scores for each expert/series presented in figure 3.26. We note that the lowest consistency with the consensus is observed for RS2 with a mean of 0.758 ± 0.148, who is rated as the most experienced for female pelvis MRI segmentation, and is the radiologist who produced 86.61% of the annotations for the FPMRId. This difference can be thus explained by RS2's approach to segmentation. The consistency of the uterus annotations is confirmed by the reported volumes, as shown in figure 3.27. Table 3.6 further supports this with low variation in

Table 3.4: Dice values for the uterus across all 10 series, where each expert is compared to the rest (i.e. (1-vs-rest): MEAN, standard deviation as STDDEV, and CV. The last column presents the overall mean. CV is a fraction.

	RJ1 vs all	RJ2 vs all	RR1 vs all	RR2 vs all	RS1 vs all	RS2 vs all	MEAN
MEAN	0.923	0.910	0.925	0.917	0.921	0.919	0.919
STDDEV	0.025	0.036	0.030	0.032	0.025	0.026	0.029
CV	0.027	0.040	0.033	0.034	0.028	0.028	0.032

the mean, standard deviation, and CV across all experts, indicating strong agreement in volume measurements. Specifically, the CV is below 10%, except for series 7, which stands at 14%. This higher variation may be due to the relatively small uterine volume in series 7 compared to the other series.

Table 3.5: Mean dice value for each expert across 10 series, when comparing expert's segmentation to the STAPLE consensus segmentation for uterus. The dice values for individual series are plotted in figure 3.26. CV is a fraction.

	RJ1	RJ2	RR1	RR2	RS1	RS2	MEAN
MEAN	0.832	0.818	0.794	0.783	0.820	0.758	0.801
STDDEV	0.084	0.067	0.108	0.104	0.085	0.148	0.101
CV	0.101	0.081	0.137	0.132	0.103	0.195	0.126

Table 3.6: Mean, standard deviation, and CV for uterus volumes in cm^3 obtained by experts for each series (s1, s2, ..., s10). CV is a fraction.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
MEAN	84.63	335.00	129.87	793.09	832.34	276.23	163.27	505.69	939.29	160.53
STDDEV	7.66	15.43	6.65	43.16	47.89	11.09	23.21	18.80	26.87	3.27
CV	0.09	0.05	0.05	0.05	0.06	0.04	0.14	0.04	0.03	0.02

Statistical Analysis. As shown in table 3.20, for the uterus there is no observed correlation between the experts' level of experience and the similarity of their segmentations to the STAPLE consensus segmentation. Further, the p-values for dice and volume scores are 0.754 and 0.993 respectively. They are above the significance threshold of 0.05, which suggests that there are no statistically significant differences between the experts' segmentations and the STAPLE consensus. Dunn-Bonferroni post-hoc test is thus not required.

Bladder

Segmentation Metrics. The evaluation of segmentation consistency for the bladder demonstrates a strong inter-expert correlation across all participating experts. As shown in table 3.7, the mean dice score across all expert comparisons is 0.880 ± 0.097, which is slightly lower than the consistency observed for the uterus (0.919 ± 0.029). This might be found surprising, given that the bladder is generally considered easier to segment than the uterus due to the better visibility. However, this lower consistency can be attributed to specific characteristics of the bladder. The bladder's contour is often not consistently defined among experts: some include the complete outer surface of the bladder in their segmentation, while others may exclude certain peripheral areas. Furthermore, because the bladder is typically homogeneous and lacks intricate internal structures, experts might devote less attention to its segmentation compared to more complex organs like the

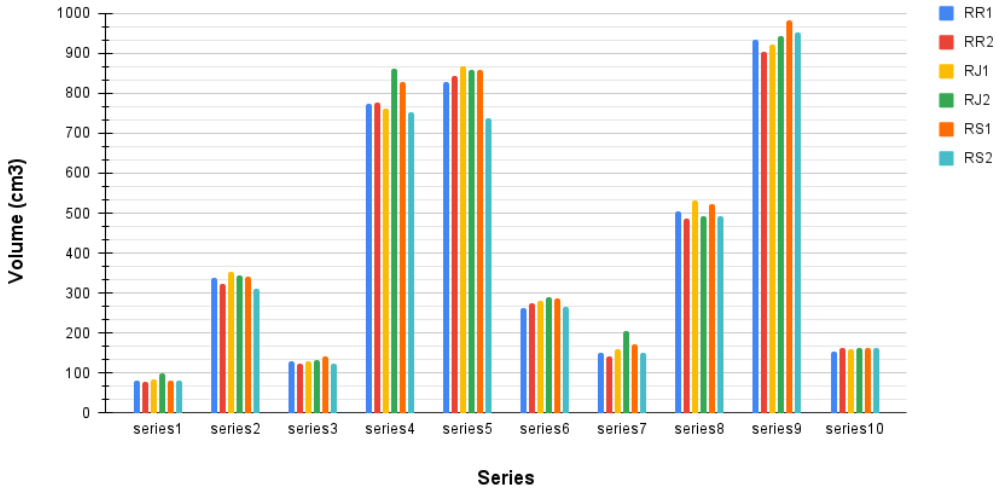


Figure 3.27: Uterus segmentation volumes for each expert and each of the 10 series.

uterus. This can lead to less precise contours and increased variability in the segmentation results. The scatter plot of bladder dice scores for each expert/series is presented in figure 3.28.

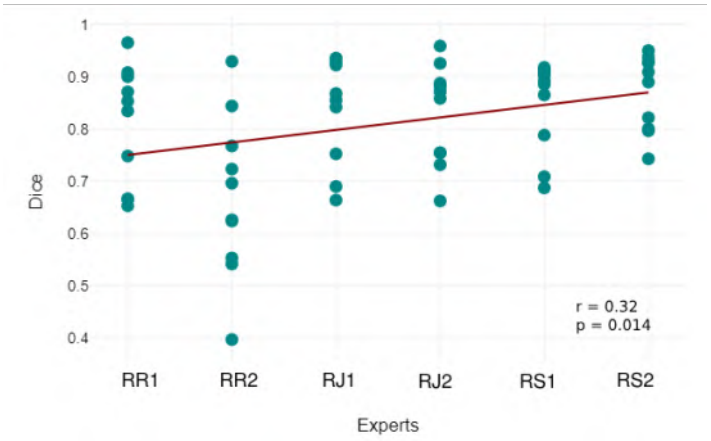


Figure 3.28: Expert's segmentation vs. the STAPLE consensus for bladder segmentation in each of the 10 series: 10 Dice scores (1 per series) for each expert. The red line represents the simple linear regression model.

When expert segmentations are compared to the STAPLE consensus segmentation, the mean dice score is slightly higher with reduced variation: 0.891 ± 0.085 (see table 3.9). Interestingly, this pattern contrasts with the results for the uterus, where experts show greater agreement with each other than with the STAPLE consensus segmentation. This difference is illustrated in figure 3.23, where the variability among experts is more evident for the bladder than for the uterus. Specifically, experts show distinct approaches to delineating the outer bladder contour, which results in inter-expert correlation being lower. Notably, expert RR2 demonstrates the lowest correlation with the STAPLE segmentation, which may be linked to their level of experience. However, when compared to the fixed STAPLE reference, these individual differences are minimised, leading to higher agreement scores.

The consistency in bladder annotations is further evaluated through the volume measurements presented in figure 3.29. Table 3.10 provides detailed statistics, showing that while most

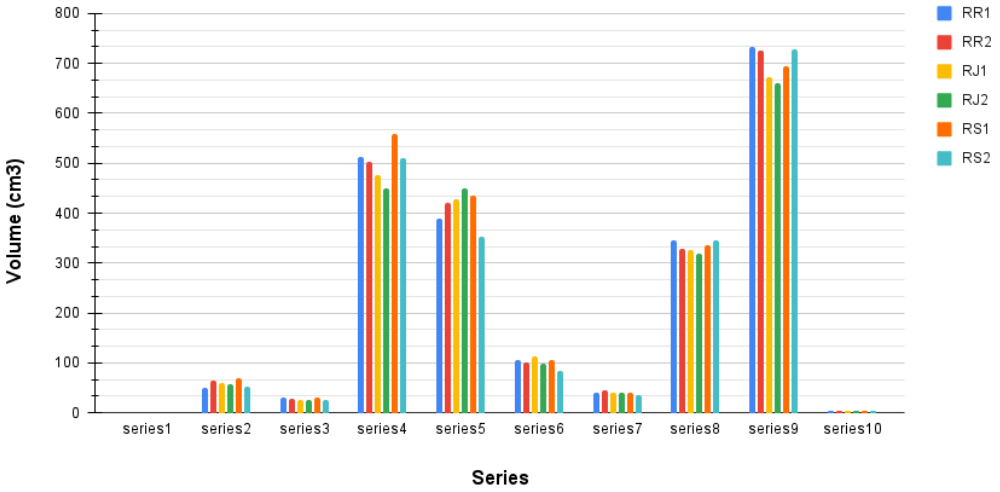


Figure 3.29: Bladder segmentation volumes for each expert and each of the 10 series.

series exhibit coefficients of variation below equal to or below 10%, indicating strong agreement, there is higher variability in certain cases. Notably, series 6 and series 10 have coefficients of variation of 55% and 19%, respectively. These elevated values may be due to the smaller bladder volumes in these series with series 6 featuring the smallest mean volume of 12.39 cm^3.

Table 3.7: Dice values for the bladder across all 10 series, where each expert is compared to the rest (i.e. (1-vs-rest): MEAN, standard deviation as STDDEV, and CV. The last column presents the overall mean. CV is a fraction.

	RJ1 vs all	RJ2 vs all	RR1 vs all	RR2 vs all	RS1 vs all	RS2 vs all	MEAN
MEAN	0.870	0.880	0.888	0.864	0.890	0.886	0.880
STDDEV	0.120	0.094	0.077	0.104	0.076	0.105	0.097
CV	0.138	0.107	0.087	0.121	0.085	0.119	0.110

Table 3.8: Mean dice value for each expert across 10 series, when comparing expert's segmentation to the STAPLE consensus segmentation for bladder. The dice values for individual series are plotted in figure 3.28. CV is a fraction.

	RJ1	RJ2	RR1	RR2	RS1	RS2	MEAN
MEAN	0.911	0.904	0.928	0.833	0.902	0.870	0.891
STDDEV	0.066	0.029	0.056	0.105	0.062	0.132	0.085
CV	0.073	0.032	0.060	0.126	0.069	0.152	0.095

Table 3.9: Comparison of expert bladder segmentations with STAPLE consensus segmentation using the dice metric.

Statistical Analysis. As shown in table 3.20, for the bladder a positive correlation between the experts' level of experience and the similarity of their segmentations to the STAPLE consensus segmentation is observed. Simply, more experienced annotators show higher similarity rate. Furthermore, the Kruskal-Wallis test yielded a p-value of 0.015 for the dice scores, indicating that there are significant statistical differences in segmentation performance among the experts. Post-hoc Dunn-Bonferroni tests identified a significant difference between experts RR1 and RR2 (adjusted p-value = 0.007), which aligns with the observations for expert RR2 based on segmen-

Table 3.10: Mean, standard deviation, and CV for bladder volumes in cm^3 obtained by experts for each series (s1, s2, ..., s10). CV is a fraction.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
MEAN	168.09	121.12	91.77	35.82	84.62	12.39	63.84	167.72	92.31	45.86
STDDEV	8.31	8.13	6.72	6.21	8.56	6.84	2.94	14.41	9.24	8.70
CV	0.05	0.07	0.07	0.17	0.10	0.55	0.05	0.09	0.10	0.19

tation metrics. Conversely, no significant differences were observed for volume measurements (p-value = 0.901), suggesting that while segmentation overlap varies between certain experts, the overall volume estimations remain consistent.

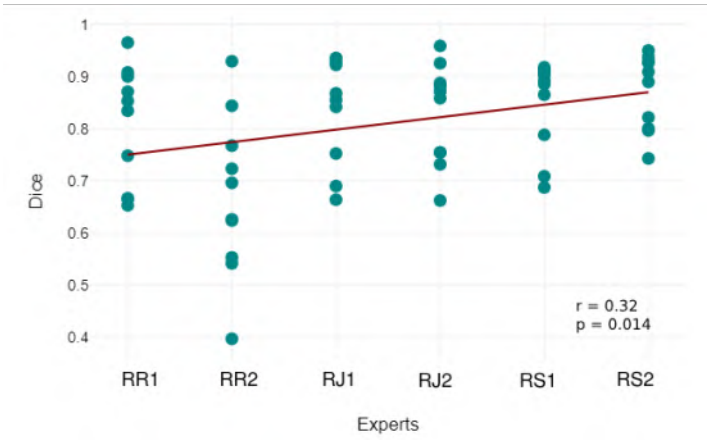


Figure 3.30: Expert's segmentation vs. the STAPLE consensus for cervix segmentation in each of the 10 series: 10 Dice scores (1 per series) for each expert. The red line represents the simple linear regression model.

Cervix

Segmentation Metrics. The assessment of segmentation consistency for the cervix revealed a moderate inter-expert agreement with notable variation among experts. The mean dice score across all expert comparisons is 0.681 ± 0.154, as reported in table 3.11. This lower correlation among experts is expected, as small-volume structures like the cervix are generally more challenging to segment accurately and tend to exhibit higher inter-expert variability. Specifically, the cervix presents two challenges in MRI segmentation, both of which make precise contour delineation difficult. Firstly, there is low MRI contrast between the cervix and the surrounding tissues. Secondly, the cervix and the uterine body display similar MRI signals, so their boundaries are primarily distinguished through morphological analysis rather than signal differences, which is more challenging. The scatter plot of cervix dice scores for each expert/series is presented in figure 3.30.

When comparing expert segmentations against the STAPLE consensus segmentation, the mean dice score is observed to be higher at 0.810 ± 0.123, as reported in Table 3.12. As with other classes, this is expected due to the nature of the comparison. We observe that expert RR2 exhibits the lowest consistency with both the other experts and the consensus segmentation, with mean dice scores of 0.631 ± 0.168 and 0.670 ± 0.156, respectively. The variability in cervix annotations is further highlighted by the reported volumes, as shown in figure 3.31. Specifically, table 3.13 shows that the coefficients of variation for cervix volumes are relatively high, ranging from 15% to 78%. As expected, the series with smaller cervix volumes, such as series 3 and 7, exhibit the highest

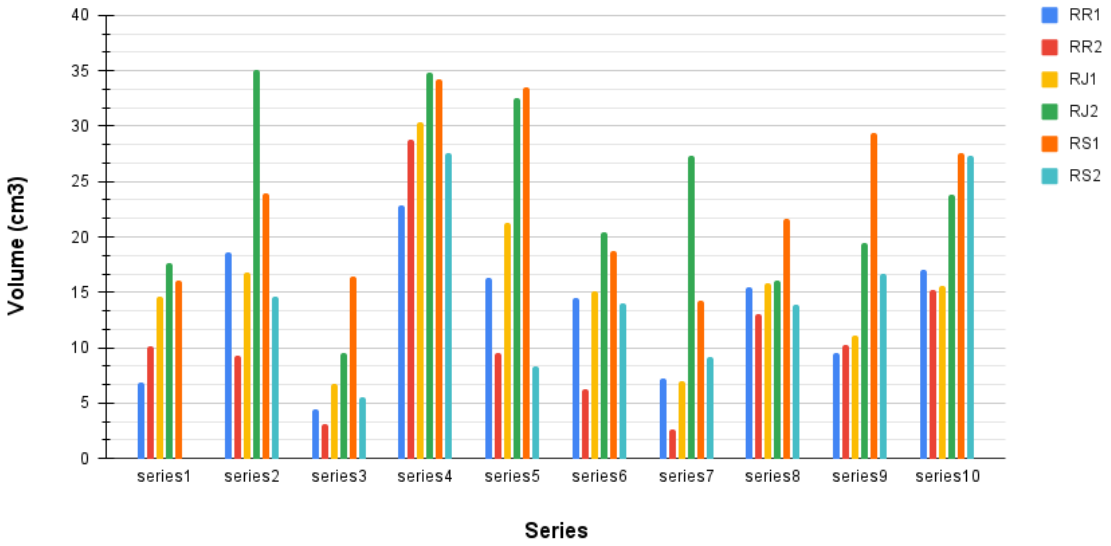


Figure 3.31: Cervix segmentation volumes for each expert and each of the 10 series.

variation.

Table 3.11: Dice values for the cervix across all 10 series, where each expert is compared to the rest (i.e. (1-vs-rest): MEAN, standard deviation as STDDEV, and CV. The last column presents the overall mean. CV is a fraction.

	RJ1 vs all	RJ2 vs all	RR1 vs all	RR2 vs all	RS1 vs all	RS2 vs all	MEAN
MEAN	0.723	0.669	0.694	0.631	0.668	0.699	0.681
STDDEV	0.130	0.164	0.142	0.168	0.169	0.141	0.154
CV	0.179	0.245	0.204	0.266	0.252	0.202	0.227

Table 3.12: Mean dice value for each expert across 10 series, when comparing expert's segmentation to the STAPLE consensus segmentation for the cervix. The dice values for individual series are plotted in figure 3.30. CV is a fraction.

	RJ1	RJ2	RR1	RR2	RS1	RS2	MEAN
MEAN	0.839	0.829	0.807	0.670	0.847	0.870	0.810
STDDEV	0.102	0.096	0.114	0.156	0.087	0.073	0.123
CV	0.122	0.116	0.142	0.233	0.103	0.084	0.151

Statistical Analysis. As reported in table 3.20 for the cervix we identified a positive correlation between the experts' level of experience and the similarity of their segmentations to the STAPLE consensus segmentation. Furthermore, the p-values of 0.034 for dice scores and 0.001 for volume measurements indicate presence of statistically significant discrepancies between experts' segmentations. Specifically, the Dunn-Bonferroni post-hoc test identified one expert pair for the dice scores: (1) RR2 and RS2. For volume measurements two expert pairs were identified: (1) RR2 and RJ2 and (2) RR2 and RS1. These findings support the segmentation metrics differences observed for the expert RR2 exhibiting lower performance as compared to other experts, which can be attributed to lower annotator's experience and the complexity of clearly delineating the cervix.

Table 3.13: Mean, standard deviation, and CV for cervix volumes in cm^3 obtained by experts across all series (s1, s2, ..., s10). CV is a fraction.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
MEAN	13.09	19.73	7.64	29.74	20.23	14.82	11.26	15.96	16.04	21.09
STDDEV	6.66	8.90	4.85	4.45	10.93	4.93	8.74	3.00	7.60	5.82
CV	0.51	0.45	0.64	0.15	0.54	0.33	0.78	0.19	0.47	0.28

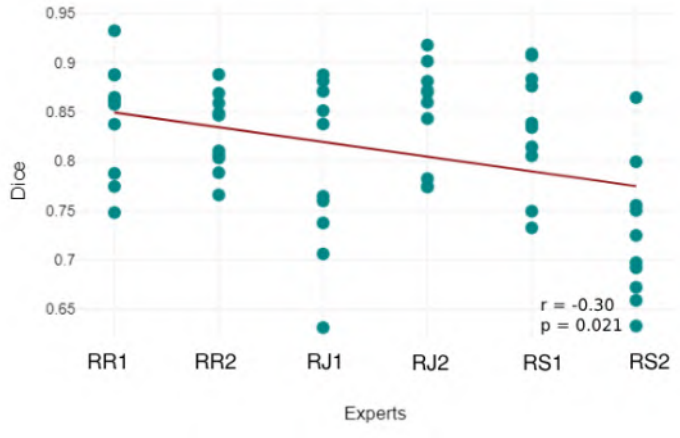


Figure 3.32: Expert's segmentation vs. the STAPLE consensus for uterine cavity segmentation for each of the 10 series: 10 Dice scores (1 per series) for each expert. The red line represents the simple linear regression model.

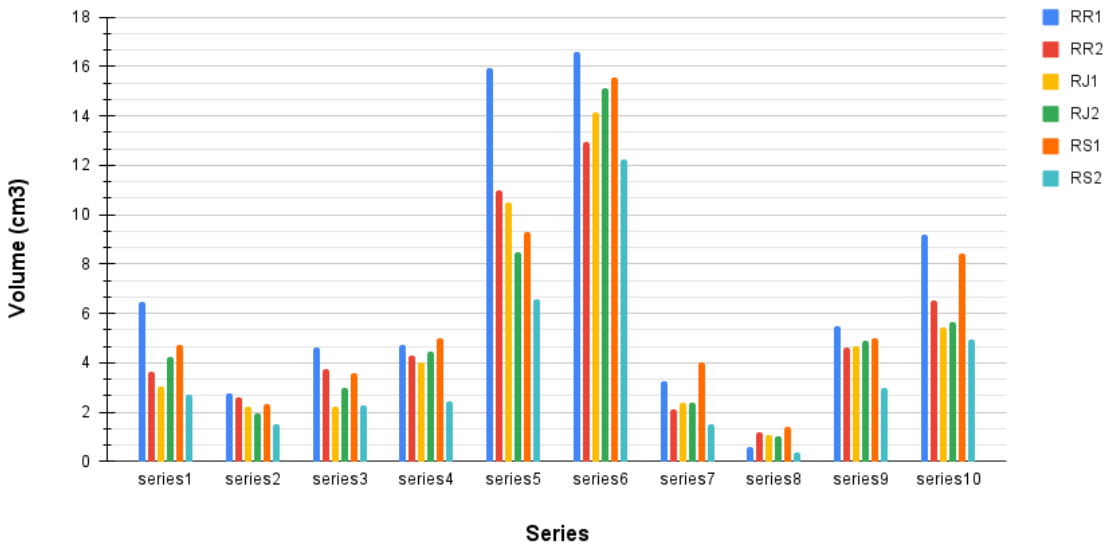


Figure 3.33: Uterine cavity segmentation volumes for each expert and each of the 10 series.

Uterine Cavity

Segmentation Metrics. The assessment of segmentation consistency for the uterine cavity shows satisfactory agreement among experts with notable variability, depending on the experts and series. The mean dice score across all expert comparisons is 0.715 ± 0.124, as reported in table 3.14. When comparing expert segmentations against the STAPLE consensus segmentation, the mean dice score is observed to be higher at 0.812 ± 0.074, which is expected. The latter is reported in table 3.15. The scatter plot of uterine cavity dice scores for each expert/series is presented in figure 3.32.

Despite the overall satisfactory mean, individual expert performances varied. RS2 shows the lowest correlation with both other experts and the STAPLE consensus segmentation, with values of 0.672 ± 0.124 and 0.725 ± 0.070, respectively. This variability is further reflected in the reported volumes, as illustrated in figure 3.33. Specifically, the coefficients of variation presented in table 3.16 are consistently high, with values ranging up to 41%, which aligns with the variability observed for the dice scores.

Table 3.14: Dice values for the uterine cavity across all 10 series, where each expert is compared to the rest (i.e. 1-vs-rest): MEAN, standard deviation as STDDEV, and CV. The last column presents the overall mean. CV is a fraction.

	RJ1 vs all	RJ2 vs all	RR1 vs all	RR2 vs all	RS1 vs all	RS2 vs all	MEAN
MEAN	0.730	0.731	0.706	0.730	0.718	0.672	0.715
STDDEV	0.124	0.119	0.122	0.136	0.117	0.124	0.124
CV	0.170	0.162	0.173	0.186	0.163	0.185	0.174

Statistical Analysis. As shown in table 3.20, a negative correlation was observed between the experts' level of experience and the similarity of their segmentations to the STAPLE consensus segmentation for the uterine cavity. Specifically, Spearman's rank correlation coefficient of -0.300 and a p-value of 0.021 were obtained. Simply, this suggests that experts with higher experience level performed worse than those with lower experience level. The Kruskal-Wallis test yielded a significant p-value of 0.006 for the dice coefficient, indicating statistically significant differences in segmentation performance among experts. Consequently, the pair RR1-RS2 was identified as a result of the Dunn-Bonferroni test, confirming RS2's lower performance as compared to other experts. This discrepancy can be attributed to three factors: (1) RS2's extensive experience contributing to FPMRId, (2) the flattened morphology of the uterine cavity, which is the smallest class in this study, and (3) the use of 5 mm slice thickness in the MRI scans, which introduces partial volume effects. Together, these factors contributed to RS2's lower performance and greater variability among segmentations between experts.

Table 3.15: Mean dice value for each expert across 10 series, when comparing expert's segmentation to the STAPLE consensus segmentation for uterine cavity. The dice values for individual series are plotted in figure 3.32. CV is a fraction.

	RJ1	RJ2	RR1	RR2	RS1	RS2	MEAN
MEAN	0.793	0.847	0.844	0.829	0.835	0.725	0.812
STDDEV	0.086	0.053	0.058	0.039	0.061	0.070	0.074
CV	0.109	0.062	0.068	0.047	0.073	0.096	0.091

Table 3.16: Mean, standard deviation, and CV for uterine cavity volume in cm^3 obtained by experts across all 10 series (s1, s2, ..., s10). CV is a fraction.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
MEAN	4.13	2.22	3.24	4.15	10.29	14.43	2.62	0.94	4.61	6.70
STDDEV	1.36	0.44	0.93	0.92	3.18	1.64	0.89	0.38	0.85	1.73
CV	0.33	0.20	0.29	0.22	0.31	0.11	0.34	0.41	0.19	0.26

Uterine Myomas

A total of 51 myomas were identified by the six experts involved in this study. However, only the myomas segmented by all participating experts were considered. Specifically, only 31 out of these 51 myomas were included in the dice and volume calculation for the following three reasons: (1) 1 myoma was noticed but forgotten to be included by one of the experts, (2) 10 myomas presented recognition challenges, and (3) 9 myomas were lobulated, leading some experts to view them as a single myoma while others saw them as multiple myomas. The number of myomas identified by each expert in each series can be seen in figure 3.21. It should be noted that the experts did not reach a consensus on the myomas present in series 1. As a result, scores for series 1 are not reported.

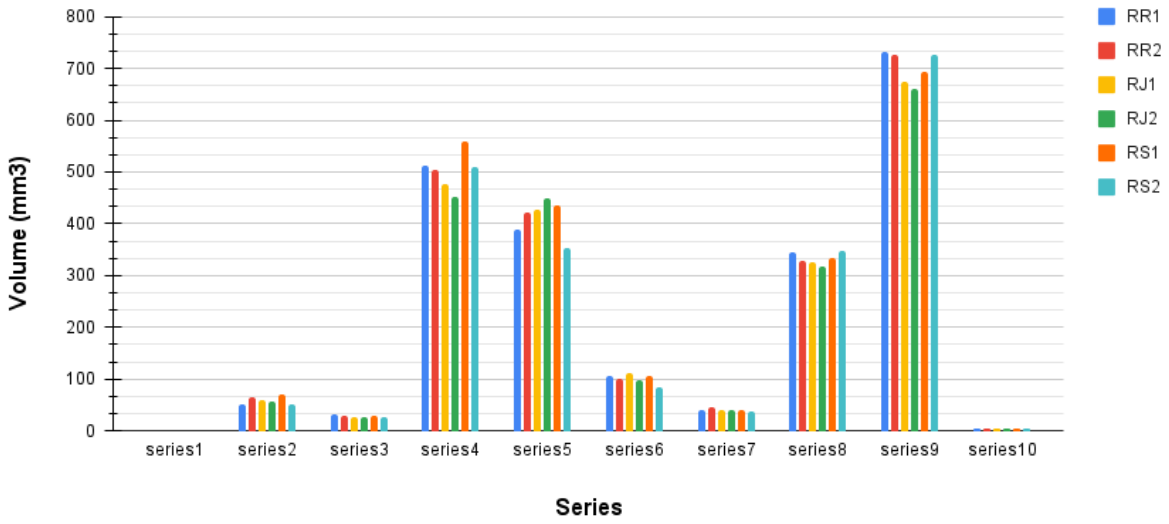


Figure 3.34: Agreed-on uterine myomas segmentation volumes for each expert and each of the 10 series.

Segmentation Metrics. The inter-expert correlation analysis for the segmentation of agreed-upon uterine myomas shows a strong level of agreement among experts. The mean dice coefficient across all expert comparisons is 0.844 ± 0.063, as shown in table 3.17.The scatter plot with dice scores for each expert/series is presented in figure 3.30. When comparing expert segmentations with the STAPLE consensus segmentation, the mean dice score is 0.836 ± 0.262, as shown in table 3.18. The standard deviation is observed to be higher compared to that of the inter-expert correlation, which can be largely attributed to recognition challenges. In particular, the presence of both small and extremely large myomas in certain series poses difficulties. Small myomas are often hard to detect, while large myomas can complicate determining the contour precisely.

To demonstrate the level of expert agreement in assessing myoma quantity, the FMA is presented in figure 3.36, showing a mean FMA of 80% among experts. However, FMA varies

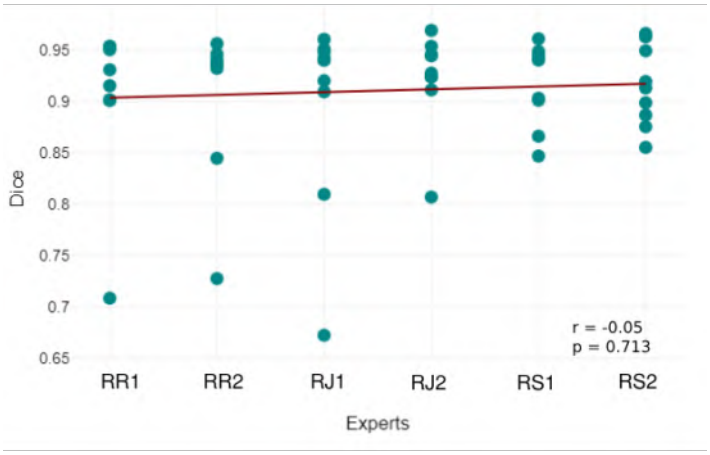


Figure 3.35: Expert's segmentation vs. the STAPLE consensus for agreed-on uterine myomas segmentations in each of the 10 series: 10 Dice scores (1 per series) for each expert. The red line represents the simple linear regression model.

significantly depending on the series. For instance, the lowest FMA is obtained for series 1 with 39%. This is due to the following two reasons: (1) small volume of the myomas described, where six are very small myomas with five estimated at less than 3 cm^3 and one around 10 cm^3, rendering them of limited clinical significance, and (2) four of the myomas being types 6 and 7 according to the FIGO, which are closely associated with intestinal loops. The reported volumes are shown in table 3.19. The low coefficients of variation, generally below 10%, confirm the strong agreement among the experts.

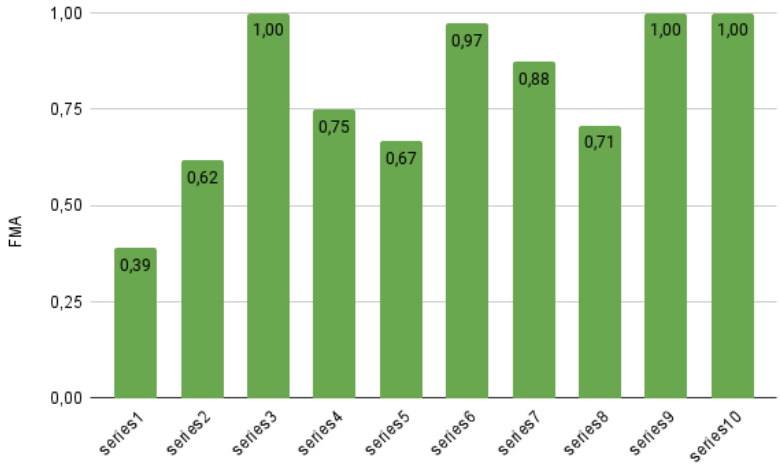


Figure 3.36: FMA for each of the 10 series.

Statistical Analysis. As presented in table 3.20, for uterine myomas there is no observed correlation between the experts' level of experience and the similarity of their segmentations to the STAPLE consensus segmentation. The Kruskal-Wallis test yielded p-values of 0.851 for the dice coefficient and 0.994 for the segmentation volume, suggesting that there are no statistically significant differences between the experts' segmentations and the STAPLE consensus. These results suggest that, despite challenges in agreeing on identified myomas in each series, the segmentation of agreed-upon myomas was consistent among experts, regardless of their experience level.

Table 3.17: Dice values for the agreed-on uterine myomas across all 10 series, where each expert is compared to the rest (i.e. (1-vs-rest): MEAN, standard deviation as STDDEV, and CV. The last column presents the overall mean. CV is a fraction.

	RJ1 vs all	RJ2 vs all	RR1 vs all	RR2 vs all	RS1 vs all	RS2 vs all	MEAN
MEAN	0.849	0.840	0.840	0.851	0.843	0.844	0.844
STDDEV	0.059	0.071	0.063	0.062	0.067	0.060	0.063
CV	0.070	0.085	0.075	0.073	0.079	0.071	0.075

3.2.3 Conclusion

In this inter-expert variability study we evaluated the consistency of manual segmentations among experts for five pelvic structures in MRI scans: (1) uterus, (2) bladder, (3) cervix, (4) uterine cavity, and (5) uterine myomas. Overall, this study demonstrates that manual segmentations among experts for pelvic MRI structures are largely consistent, suggesting that annotations can be reliably performed by different annotators. However, specific series and cases may exhibit significant discrepancies due to their unique characteristics especially common in female pelvis MRI. This means that each series should be evaluated and validated on an individual basis during the annotation process. In the following, we detail the main observations.

Table 3.18: Mean dice value for each expert across 10 series, when comparing expert's segmentation to the STAPLE consensus segmentation for agreed-on uterine myomas. The dice values for individual series are plotted in figure 3.35. CV is a fraction.

	RJ1	RJ2	RR1	RR2	RS1	RS2	MEAN
MEAN	0.805	0.931	0.817	0.816	0.826	0.823	0.836
STDDEV	0.297	0.051	0.296	0.295	0.293	0.292	0.262
CV	0.369	0.054	0.363	0.362	0.354	0.354	0.313

High inter-expert agreement was achieved for larger and more distinct structures like the uterus and bladder, with excellent correlations for uterine volume and very satisfactory correlations for bladder volume and myomas. These classes exhibited high mean dice scores (above 0.840), indicating strong consistency among experts regardless of individual approaches or levels of experience. This suggests that large objects of interest, well-defined anatomical boundaries and higher MRI contrast facilitate more uniform segmentations.

Table 3.19: Mean, standard deviation, and CV for agreed-on uterine myomas' volumes in cm^3 obtained by experts across all series (s1, s2, ..., s10). Series 1 is marked as NA due to the absence of agreement between the experts.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
MEAN	NA	58.97	28.05	501.79	413.00	101.22	40.59	333.37	702.15	4.27
STDDEV	NA	7.07	2.54	36.64	35.23	9.69	2.98	11.24	30.85	0.40
CV	NA	0.12	0.09	0.07	0.09	0.10	0.07	0.03	0.04	0.09

In contrast, smaller and more complex structures such as the cervix and uterine cavity showed moderate agreement, with satisfactory correlations for the uterine cavity and moderate for the cervix. Challenges in segmenting these areas stem from factors like low contrast with surrounding tissues, intricate morphology, and the impact of partial volume effects due to the use of thick-slice MRI sequences (5 mm in this study). These complexities contribute to greater discrepancies in expert annotations, highlighting that the cervix and uterine cavity segmentation would benefit

from special attention and higher scan quality. At the same time, the segmentation of uterine myomas, despite initial challenges in consensus on myoma identification, resulted in high agreement for the agreed-upon myomas. This indicates that while certain myomas are missed by experts, agreed-on myomas are consistently delineated.

Table 3.20: Results of the statistical analysis for all classes. Values indicating significant statistical differences are highlighted in bold. The last column lists the identified pairs exhibiting these differences, separately for dice and volume metrics.

Class/Result	**Spearman**		**Kruskal-Wallis**		**Dunn-Bonferroni**	
	r_s	p	p (dice)	p (volume)	Dice	Volume
Uterus	0.010	0.949	0.754	0.993	NA	NA
Bladder	0.320	**0.014**	**0.015**	0.901	RR1 - RR2	NA
Cervix	0.320	**0.013**	**0.034**	**0.001**	RR2 - RS2	RR2 - RJ2, RR2 - RS1
Uterine Cavity	-0.300	**0.021**	**0.006**	0.487	RR1 - RS2	NA
Uterine Myomas	-0.050	0.713	0.851	0.994	NA	NA

The statistical analysis highlighted that the influence of expert experience on segmentation performance is not uniform across all classes. For certain structures, a positive correlation between experience level and segmentation agreement was observed, while for others, no significant correlation or even a negative correlation was found. This can be explained in certain situations. For example, extensive prior experience of RS2 in FPMRId or lower experience levels of RR1 and RR2 result in expected lower correlation with other experts. Additionally, the use of the STAPLE algorithm to define the consensus among experts proved effective. This led to its adoption in other contexts requiring consensus on varied annotations, such as in endometriosis laparoscopic surgery (The European Parliament and the Council of the European Union, 2024a). Specifically, this type of surgery lacks profound standardisation, with significant variability in surgeons' annotations due to individual approaches. However, the algorithm remains limited due to its tendency to potentially disregard accurate but minority segmentations and reliance solely on consensus without accounting for anatomical continuity. This suggests that it should be used along other approaches.

Chapter 4

Interactive Neural Segmentation

4.1 Introduction

Image segmentation is an essential component of many visual processing systems, which involves classifying each pixel or, equivalently, delineating the regions containing pixels of the same class. In medical image analysis, the images are often patient scans from modalities such as MRI or CT. MRI segmentation is a tremendously difficult task, owing to it being 3D, low contrast, noisy, low resolution and artefacted. Existing segmentation approaches can be divided into three settings based on user involvement: manual, automatic and interactive. The manual approach is the most time-consuming, as each pixel has to be attributed a label independently, which may require hours for a single MRI. It is error-prone and infeasible in the clinical environment. At the other extreme lies the automatic approach, which works without user involvement. This strongly limits its applicability, as a clinician operator shall validate and possibly edit the result before its use in a therapeutic act. The interactive approach trades-off manual and automatic features: it typically involves an automatic part with an extent of user control. Both aspects are crucial for systems designed for the clinical environment, where there generally are three main constraints: (1) decision-making should be human-controlled, (2) time is limited, and (3) high accuracy is desired. Creating interactive systems addressing these three concerns is therefore essential to simplify, speed up and secure segmentation in the clinical environment.

The automatic approach is largely dominated by DL, which overturned classical methods over the last decade in many segmentation tasks (Cardenas et al., 2019; O'Mahony et al., 2020). In contrast, interactive DL methods present specific difficulties and have yet received relatively limited attention (Ramadan et al., 2020). Concretely, DL interactive segmentation requires embedding a network in an interactive-loop system allowing the user to interact. Indeed, the network inputs must include the user feedback, which depends on the network outputs. This creates a dependency between the inputs and outputs of the network, which is poorly resolved by a regular training process from static data. Specifically, the input configuration and training process of interactive existing DL methods do not reflect how the user interactions are provided at test time. They consequently do not take full advantage of having user interactions as input, missing two key aspects: (a) realistic interaction simulation - real interactions are positioned rationally, but often scarcely and randomly distributed, an aspect which is not modelled in existing simulation approaches for training; (b) temporal interaction information - inherently present at all times in the real world, but overlooked by the existing interactive segmentation methods.

Dynamics or temporal information are additional cues typically used in video segmentation and tracking methods, which take advantage of the order and similarity of adjacent video frames. In interactive segmentation, a user interacts depending on the current segmentation result they observe, which is conditioned by both their interactions and the system's result so far. Hence, the ordering of interactions is highly important and should not be altered, as they otherwise become less informative. Intuitively, capturing the interaction order should be beneficial in any interactive framework, including interactive segmentation.

We propose a general DL interactive segmentation framework and training methods for multi-class semantic instance segmentation. Our system consists of an embedded network, a user interaction loop and an interaction memory. First, the user reviews the current segmentation result and, if satisfied, accepts. Otherwise, the user may quickly make simple corrections by placing points or strokes to refine the segmentation, which is achieved by a special input configuration of the embedded network. Indeed, this network inputs the image, user correction masks, and possibly other memorised parameters, and outputs the segmentation probability maps. The system then loops back to the user review step, whilst updating the interaction memory to keep track of the user corrections throughout the interactions.

In practice, the additional temporal information is represented by a neural network input structured differently than existing work. Existing works store all the interactions in the same mask, discarding the order of the interactions and hence the temporal information. We call such input structures Cumulative Interaction Memory (CIM). In contrast, we propose Sequential Interaction Memory (SIM), which stores a sequence of states instead, where each state is a pair of user input and corresponding segmentation output. Simply put, SIM is a sequence of ordered user actions and their results in time and carries temporal information by definition. The proposed architecture takes an image and a SIM as inputs and produces a segmentation as output. The system then adds this segmentation along with the latest user interaction mask to the SIM and proceeds to the next interaction round. In practice, SIM is represented by a tensor of a certain size, depending on the memory size, and is used as an input to the network at all times.

Our contributions are threefold. First, we propose a general DL-based interactive multi-class semantic image segmentation framework with a user interaction loop. Second, we propose a sequential interaction memory, which keeps track of the segmentation results and user corrections, maintaining sequentiality within the system. Third, we propose a general dynamic data training process, which simulates the correction-focused and sequential nature of human user feedback by learning from interaction sequences of a virtual user and minimises interaction-dependence, improving performance.

We demonstrate our framework in three tasks. The first task is multi-class semantic MRI segmentation of the female pelvis, for which we created a new dataset collected in our hospital. We validate the results against automatic and existing interactive systems with the standard metrics and perform an ablation study of our system's components. We report results of a user study with 8 experts conducted with both senior and junior medical users in terms of both standard metrics and elapsed time, using a specifically developed graphical user interface connected to our system. We also study the influence of the number of provided user interactions on the framework's performance, including when using the framework in the automatic mode with 0 clicks provided. The second and third tasks are respectively the multi-class semantic liver and pancreas CT segmentation, using the 'Liver Tumours' and 'Pancreas Tumour' medical segmentation decathlon

datasets (Simpson et al., 2019). We validate the results against automatic approaches participating in the ongoing medical segmentation decathlon challenge (Antonelli et al., 2022). For these tasks, we instantiate our system with an existing encoder-decoder architecture optionally featuring RNN (Rumelhart et al., 1986) modules.

4.2 Related Work

We review classical and DL approaches to medical image segmentation, distinguishing automatic and interactive approaches for each.

Classical automatic segmentation encompasses a wide variety of methods (Zhu et al., 2016). Their performance is usually insufficient to achieve clinically-acceptable accuracy and they have been largely taken over by DL in many tasks (Cardenas et al., 2019). In contrast, classical interactive methods are still widely used. The most well known ones are probably the Graph Cuts (Boykov and Jolly, 2001), Random Walker (Grady, 2006b) and GEOS (Criminisi et al., 2008). They achieve acceptable performance for simple cases. However, medical data often feature structures with complex shapes and poorly defined contours, noise and artefacts. This results in a substantial increase of user time required to perform segmentation and limited achievable accuracy.

DL-based automatic segmentation includes a multitude of methods. A review and evaluation of over 100 methods (Minaee et al., 2021) was conducted with ResNet (He et al., 2016) extensively used as a backbone, represented by EMANet (Li et al., 2019). It achieved top scores on the PASCAL VOC dataset together with (Zoph et al., 2020), which adopts NAS-FPN (Ghiasi et al., 2019) with EfficientNet-L2 (Xie et al., 2020). Most of the models use an encoder-decoder architecture (Minaee et al., 2021). This includes the U-Net (Ronneberger et al., 2015), with a wide spectrum of applications (Siddique et al., 2021), and recent variants (Futrega et al., 2021; Siddiquee and Myronenko, 2021) reaching top positions in the BraTS challenge 2021. Automatic MRI segmentation was attempted for various targets, including the kidney (Kline et al., 2017), the prostate (Guo et al., 2016) and brain tumours (Havaei et al., 2017). These methods demonstrate state-of-the-art performance in their respective tasks. However, they are automatic and do not allow the user to interact. Automatic segmentation is highly appropriate in applications which cannot involve user interactions in essence, such as real-time organ tracking. In contrast, many applications require validation and corrections from a certified user. For such applications, the direct use of automatic DL methods is inappropriate.

The integration of DL within interactive segmentation systems is a major challenge. A simple approach is to use a classical interactive method to post-process the result from an automatic DL method (Wang et al., 2018) or correct it manually (Shan et al., 2020). Such systems inherit the intrinsic limitations of the chosen classical method. A more advanced approach is to use a neural network to process user feedback in an interactive-loop system (Vrooman et al., 2006; Wang et al., 2019a; Zhou et al., 2019, 2022; Liao et al., 2020; Sakinis et al., 2019; Jahanifar et al., 2021). These methods use a network which takes the image and user interaction masks as inputs. Training is challenging owing to the loop. Existing approaches generate user interaction masks from labelled data, either statically before training or dynamically during training, or attempt to avoid training altogether. Static data training methods (Wang et al., 2019a; Zhou et al., 2019, 2022) limit the system's generalisation and interaction effectiveness. Intuitively, a real user interacts based on the current segmentation they observe. In other words, the goal of the user is to improve upon what

is already there. Hence, it is sound that mimicking this mechanism of acting sequentially is more faithful and true-to-practice than the previous mechanism, namely Static Data Generation (SDG), not taking past segmentations into account.

Dynamic data training methods (Vrooman et al., 2006; Liao et al., 2020; Sofiiuk et al., 2021; Jahanifar et al., 2021; Koohbanani et al., 2020) mimic this mechanism and simulate user interactions by sampling missegmented regions. This is done once from a single prediction (Vrooman et al., 2006) or from the latest segmentation result (Liao et al., 2020; Sofiiuk et al., 2021). Usually, such methods rely on a virtual user, which generates user input artificially at training time, since the involvement of real users is not feasible. These methods diversify the training data and improve performance. However, previous works using dynamic data training have two shortcomings: first, they consider only individual classes for click placement, which is not well-adapted to the medical scan data naturally containing multi-instance or multi-component structures, and second, they do not handle multi-class multi-label multi-instance problems with multiple components per class. These problems make medical scan segmentation challenging, as they incur the fragmentation of classes into multiple components, all compounded by the inherent noise, variability and complexity of medical image scenes. In order to exemplify their terms, consider for instance, the female pelvis MRI dataset we assembled. It has multiple properties typical for medical scan datasets, namely (1) multi-class - the dataset contains multiple classes (that is, `uterus`, `bladder`, `tumour` and `cavity`); (2) multi-label - certain classes overlap (e.g. `uterus` contains `tumour` and `cavity`); (3) multi-instance - certain classes contain multiple instances (there can be multiple tumours per image); (4) multi-component - an instance of each class in the image might be split into multiple closed contours due to medical scan slicing and the shape of the object in question.

Alternatively, training-less methods were proposed to bypass the training challenges (Jang and Kim, 2019; Sofiiuk et al., 2020). Specifically, they use an automatic segmentation network interactively via inference-time optimisation and improve performance. However, these methods have certain drawbacks. First, they require backward passes using gradients, leading to a computational overhead. Second, their applicability is limited because widely used frameworks often lack support for the backward passes on mobile devices. These two factors make it difficult to apply them in practice, provided the limited availability of the high-performance GPUs in clinical workstations and laptops. An open-source interactive segmentation platform (Diaz-Pinto et al., 2022) was recently made available, which offers both DL-based (Wang et al., 2018; Sakinis et al., 2019) and classical methods (Boykov and Jolly, 2001), inheriting their limitations.

The existing methods do not reproduce the typical sequentiality of real user interactions. The lack of sequentiality is a consequence of the interaction memory used in these systems, which simply accumulates the user corrections, discarding ordering. In contrast, we argue that the order of the user corrections can be directly used for training and lead to performance improvements. In short, the rationale is that the order in which the user corrects the segmentation in an interactive system depends on the current segmentation estimate. The order of interactions can thus not be changed and forms an important piece of information to the system. A sequential memory was used in (Zhou et al., 2022) to 'transfer' the user interaction recorded on one slice to the other slices, but was not used to exploit sequentiality during slice segmentation.

In contrast to existing work, our framework uses a sequential interaction memory which captures the sequentiality of user interactions at training and inference times. Furthermore, the proposed framework does not require specific modifications for inference and preserves low inference

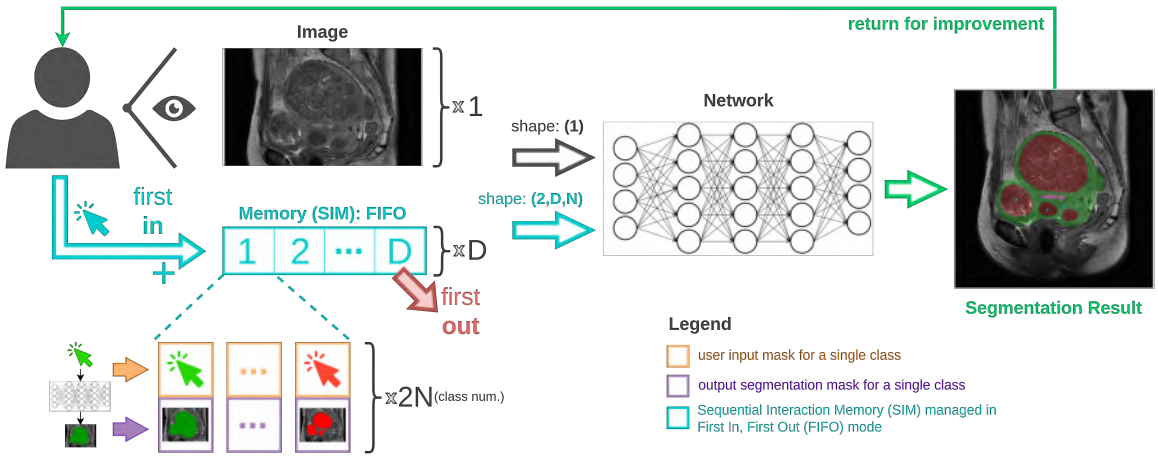


Figure 4.1: Proposed interactive system, featuring a network embedded in a user interaction loop and an interaction memory.

time. Additionally, the proposed dynamic data training specifically targets higher automatisation and generalisation at testing time by introducing a set of rules allowing for extreme variability of simulated inputs.

4.3 Applicative Scope

While our framework may be applied to numerous segmentation problems, we focus on the interactive slice-by-slice female pelvis MRI segmentation, involving five classes: `uterus`, `bladder`, uterine `cavity`, `tumour` and `background`. The intended use is surgical planning and surgical augmented reality (Collins et al., 2020). We created a female pelvis MRI dataset, consisting of 97 MRI series with 3066 slices in total, manually annotated in 3D Slicer (Kikinis et al., 2013) and in MITK (Goch et al., 2017) by expert radiologists. This took from 10' to 50' per series with 25' on average with certain series (for instance with strong uterus deformation as in (4) in figure 1.15) taking more than 1 hour, which is clearly infeasible in the clinical setting. The segmentation of anatomical structures of the female pelvis is particularly challenging due to a large variance in their representation, including shape, size, position, orientation and texture among the patients, with and without pathologies. Moreover, it is typical for MRI data to suffer from non-uniformities of the low frequency intensity areas, which is detrimental to the network learning capabilities. Difficult samples can be seen in figure 1.15. On top of that, the target anatomical structures form a naturally imbalanced dataset, where `background` takes 96.15%, `uterus` 2.11%, `bladder` 1.02%, `tumour` 0.67% and uterine `cavity` 0.05%. The strongest imbalance is observed for uterine `cavity` and `background`, whose average ratio of volumes is 0.057%. The classes are also unevenly distributed throughout the dataset due to the number of the tumours varying among the series between 0 and 27. These factors further complicate learning and generally result in much lower performance on smaller classes if no mitigation against class imbalance is introduced. Our objective is to develop a segmentation system which minimises the time required to complete the segmentation with acceptable accuracy, while allowing an expert reviewer to have control and guide the segmentation, as and when necessary.

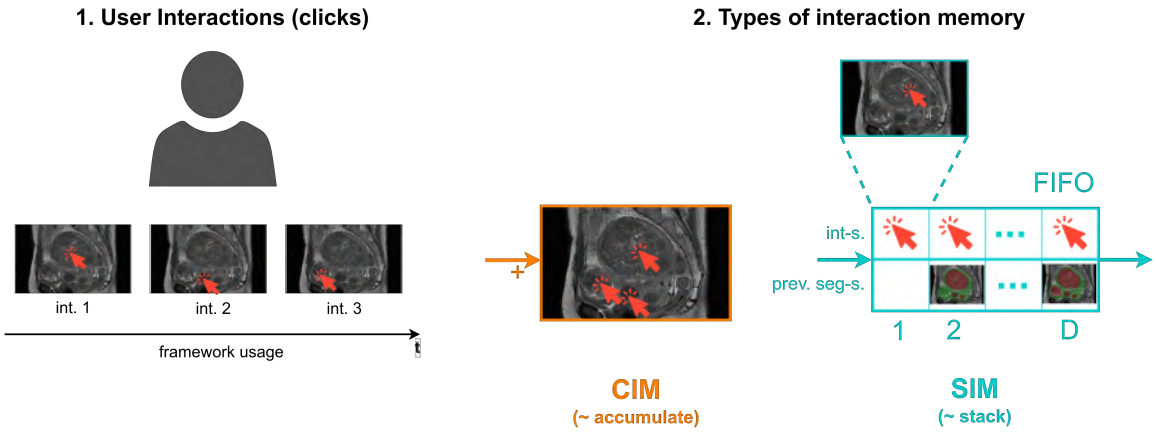


Figure 4.2: Interaction memory differences: (1) three individual interactions provided one-by-one with respective intermediate segmentation results obtained; (2) the CIM and SIM are shown, which memorise the interactions from (1).

4.4 Methodology

We describe the system and then the training process.

4.4.1 System

We give the system's general structure and then the internal memory's structure.

Structure. We build the proposed system shown in figure 4.1 starting with a basic interactive segmentation system named `base`, featuring an interaction loop. This system does not have a memory of user corrections or previous segmentation results and processes each set of user corrections in isolation. The interaction loop allows iterative refinement by forming new inputs through a combination of network outputs and user corrections. The system is generic as it does not depend on a specific network architecture, as long as the network takes both the image and the user corrections as inputs. The user corrections are represented by N binary masks, where N is the number of classes. The network inputs are concatenated into a single tensor of size $H \times W \times C$, where $H \times W$ is the image size and C is the number of channels, varying depending on the system. For the base system $C_{\text{base}} = 1 + N$. Indeed, as there is no memory in this system, the network takes the image as the first channel and the binary masks of the user corrections for the N classes as the next N channels. This strongly harms user experience as the past user corrections are forgotten by the system at the next interaction (Wang et al., 2019a, 2018).

Cumulative and Sequential Interaction Memory. We introduce an interaction memory, whose role is to keep track of user corrections. For that, we define a system state as a combination of user corrections and the corresponding network outputs. For the task of multi-class segmentation, a single state consists of a probability map for the network outputs and a binary mask for the user corrections, for each of the N classes. It is important to make a distinction between the interaction memory and the internal memory found in the RNN. The interaction memory tracks and stores system states, represented by inputs and outputs of the network. Indeed, the interaction memory is external to the network and does not depend on a specific network architecture. The RNN memory, however, is internal and specific to the network architecture, enabled by passing hidden

states from step to step and represented by weights.

Existing works use an interaction memory, which aggregates the system states by merging the successive interaction masks (Amrehn et al., 2017; Zhou et al., 2019; Liao et al., 2020). We call this a CIM. The network takes the image and the merged user correction masks, and its input tensor thus has $C_{\text{cim}} = C_{\text{base}} = 1 + N$ channels. This type of memory discards the ordering of interactions - the sequentiality, typical of user corrections. We introduce a second type of interaction memory which, in contrast to CIM, preserves the past D system states, hence the user's sequential behaviour. We call this a SIM, and the number of states D the SIM's size or depth. The network takes an image and the SIM as inputs, which are combined to form the input tensor with $C_{\text{sim}} = 1+2DN$ channels. The factor 2 comes from each state containing both N interaction masks and N probability maps of intermediate segmentation results. Simply put, SIM is a container for naturally ordered input-output pairs both at training and at testing times. In other words, it is a representation of the temporal information associated with user inputs. The general differences between CIM and SIM are schematically shown in figure 4.2.

We note that the SIM does not change the system's applicability, which remains generic with respect to the data type and embedded network architecture. In our ablation study we show that RNN's suitability for sequential data may further reinforce the proposed framework.

4.4.2 Training with Dynamic Data Generation

In an interactive-loop system with an embedded network, the inputs depend on the outputs. This means that a regular training process from static data will poorly reproduce the real system usage at test time, limiting the achievable accuracy and user interaction efficiency. To resolve this, we propose a dynamic training approach, where the training data is generated from the labelled dataset during training by a virtual user. The basic idea of the virtual user is to generate a set of corrections similarly to a real user, whose involvement in training is not feasible. These corrections are represented by one binary mask per class, populated by foreground clicks for each class, including the `background` class. The click is handled by an interaction-control process, which exploits the difference image between the latest network output and the ground truth. This difference image gives a set of mislabelled regions, containing both under- and over-segmented regions. The position of the click is chosen randomly in the largest region, following a probability map whose maximum is at the region centre, decreasing towards the region boundary and vanishing outside the region. A general schematic of the Dynamic Data Generation (DDG) process can be seen in figure 4.3. It shows an example of interaction generation for a single image containing 4 `tumour` instances with a single click generated per interaction round. In practice, the process seen in figure 4.3 is applied online during training for each image in the batch before passing on to the next batch. The standard training routine where batches are processed one-by-one is not changed, neither is any preprocessing done before the training process. Simply, compared to standard training, there is only an additional interaction generation routine for each image, similar to how online data augmentation is done.

In a typical segmentation task, each class may be represented by multiple individual components. Recall that a component is a set of spatially connected pixels pertaining to the same class in the image. When applied to our task of FPMRI segmentation, this frequently occurs for all classes due to the presence of multiple instances of the same class (for the tumours) and due to the nature

of the 3D MRI volume slicing (for the bladder, the uterus and the uterine cavity). For example, in certain cases the uterus' cross section may be represented in the image by multiple components due to its shape. We address this by changing how the clicks for each image are simulated and split the click simulation process in two steps. In step (a), the virtual user exceptionally considers each component of each class for a potential location. In step (b), the virtual user considers the mislabelled regions with larger size having higher probability of a click to be added.

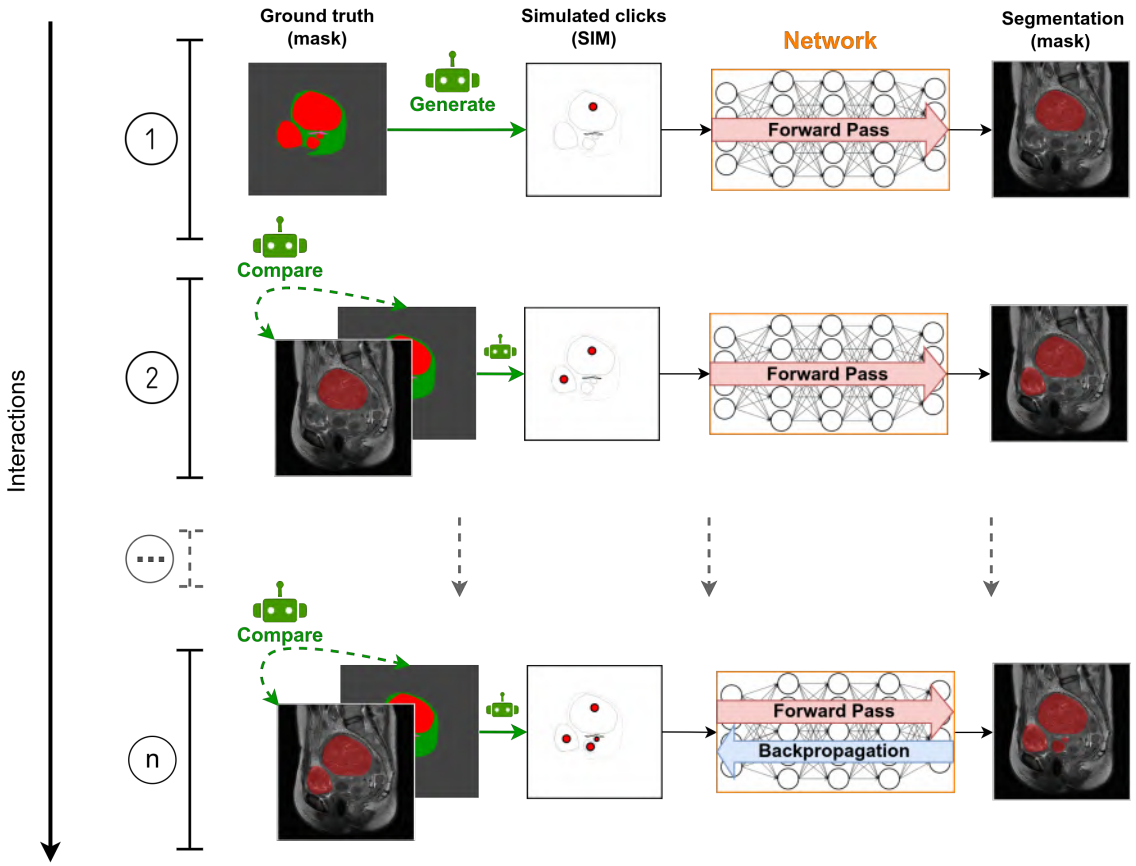


Figure 4.3: An example of DDG for a single input image. The schematic is read from left to right and top to bottom i.e. row by row. DDG is applied to each image each time it is encountered in the dataset. Precisely, DDG simulates a virtual user to generate the maximum of n interactions for a component of a single class. The class is represented here by 4 `tumour` instances (in red). At each interaction round, inference is performed to obtain an intermediate segmentation result, which is then compared with the ground truth to generate a new interaction based on their discrepancy. Clicks at previous interaction rounds are stored in SIM and carried over to the next round. Backpropagation is performed when all n interactions were simulated. The actions of the virtual user are marked in green. DDG is applied to all classes simultaneously following the rules in section 4.4.2.

In addition to interaction placement, our system implements an interaction-independence scheme, designed to ensure robustness against imperfect user behaviour at test time, with the following four main rules:

1. The maximum number of simulated interactions per component of each class is limited, typically to 3. The minimum is 0.

2. The probability of adding a subsequent interaction starts at $p \leftarrow 1$ and linearly decreases as $p \leftarrow p - \frac{1}{t}$ after each interaction round, where t is the maximum number of training interactions.

3. At each image, a random class is selected for which the user interactions are not generated.

4. A percentage of all generated interactions is held out. We typically use 80%.

These rules, along with the interaction placement control, allow the system to generate sufficiently varied interaction data throughout the training process and decrease the system's reliance on interaction supply. Specifically, step (a), as well as rules 3 and 4 do not exist in previous work. They ensure a high level of variety in the generated data and significantly reduce interaction dependence, as evidenced by the experiments in section 4.5.2. The rationale for rules 3 and 4 is threefold: 1) the framework should produce annotations for the classes not explicitly clicked on, 2) the network should consider image features instead of relying solely on user interactions and 3) the framework should be capable of automatic segmentation with no interactions provided.

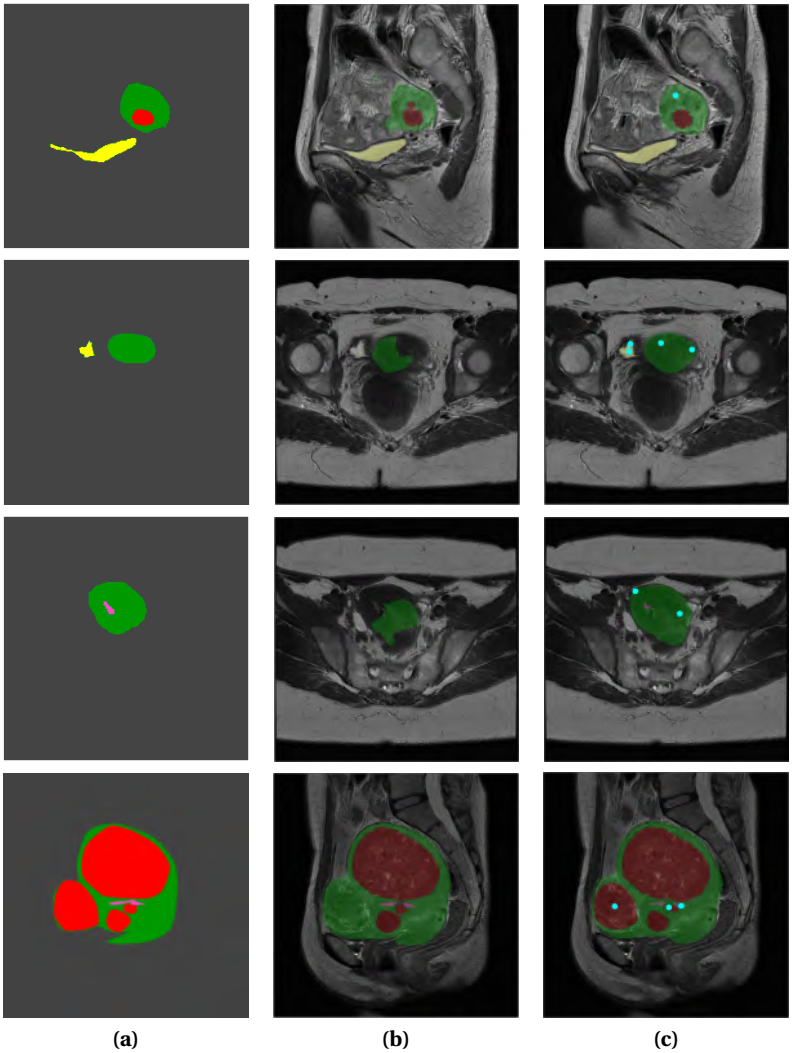


Figure 4.4: Segmentation results, where `uterus` - green, `bladder` - yellow, `tumour` - red, `cavity` - pink and user clicks - cyan: **(a)** ground truth; **(b)** `Auto`; **(c)** human user-controlled `DDG-SIM`.

Training with the proposed SIM means filling its D states with realistic data produced by the virtual user. Specifically, DDG is the method used to form a virtual user, which generates user input artificially at training time, since the involvement of real users is not feasible. Therefore, DDG is used to fill in the sequential interaction memory during training. We thus run the system for D iterations with fixed weights to populate the SIM with simulated user input data prior to back-

propagation. This is done anew each time the image is encountered in the dataset, similarly to classical data augmentation. We choose D experimentally with the goal of maximising the performance with the minimum number of interactions. At the same time, any or all of the D states may remain empty both at training and testing time to obtain a fully automatic segmentation result to be validated or subsequently refined. The DDG routine is given below as pseudo-code applicable to one specific sample image:

1. **Input** click probability p, maximum number of training interactions t
2. If $p = 1$, simulate an initial click for each component
3. If $p < 1$, simulate a corrective click for each class for the largest mislabelled region with probability p
4. Update p as $p \leftarrow p - \frac{1}{t}$
5. (rule 3) Randomly choose a class and ignore its simulated clicks
6. (rule 4) Ignore 80% of all simulated clicks
7. Form the interaction mask $\mathscr{M}$ from the simulated clicks
8. **Output** click probability p, interaction mask $\mathscr{M}$

The click probability p is managed for each image independently. It is initially set to 1 and then updated by the DDG routine.

4.5 Experimental Results

We describe the experiments and report the obtained results, which are then discussed in section 4.6.

4.5.1 Experimental Setup

We give implementation details and describe data augmentation and training.

Implementation. The proposed framework and methods are not tied to a specific network architecture. We instantiate our system with an existing encoder-decoder architecture featuring RNN modules, also called AlbuNet (Shvets et al., 2018), optionally modified with LSTM layers in the decoder. Specifically, we use a ResNet34 (He et al., 2016) encoder and a decoder equipped with a standard convolutional layer and a matching convolutional Long Short-Term Memory (LSTM) layer at every step of the upsampling path as shown in figure 4.5. The reason for which we chose this U-Net is its known efficiency in the field of medical image analysis, as shown in (Chaisangmongkon et al., 2021; Kusakunniran et al., 2023). The choice of the encoder follows the same principle. However, our framework is flexible as it allows for the use of various base architectures that can accommodate an additional temporal dimension in the input image, such as a different UNet or, for example, DeepLab v3 (Chen et al., 2017). This adaptability is a strength of our framework.

LSTMs are generally effective at processing sequences of data due to cells containing input, output and forget gates. A typical input for an LSTM network is sequential data where the order

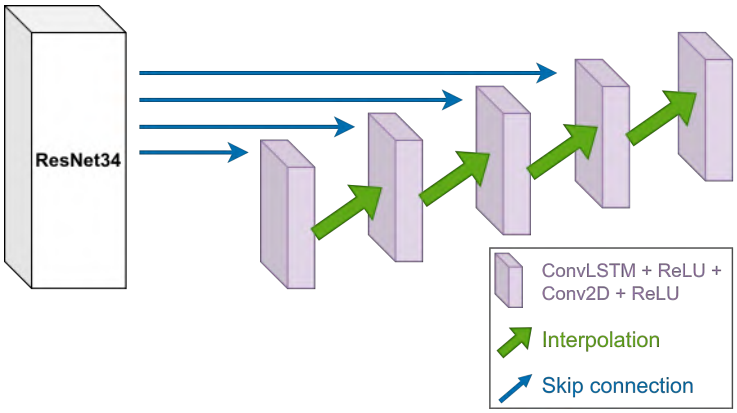


Figure 4.5: A general schematic of the network architecture used in the complete proposed system (DDG-SIM): the ResNet34 encoder pre-trained on ImageNet and a decoder with a convolutional LSTM layer at every step of the upsampling path.

and timing of individual elements are significant. This type of data is characterized by its temporal or sequential nature, meaning that the relationship between elements depends on their position in the sequence. These properties make LSTMs beneficial for our framework, where LSTM layers reinforce sequentiality by retaining and reusing useful information about previous interactions, and improve performance, as shown by the ablation study in section 4.5.2.

As compared to CIM, for which the network's input tensor has $C_{\text{cim}} = 1 + N$ channels, where N is the number of classes, with SIM, we have $C_{\text{sim}} = 1 + 2DN$ channels, where D is the SIM's size or 'depth'. The first channel is the image. The factor 2 comes from each of the D states containing both N interaction masks and N probability maps of intermediate segmentation results. For an LSTM, the input data shape could be represented as a triplet 'samples, time steps, features', which aligns well with SIM as the samples are taken as the $2N$ masks, the time steps as the D states and the features as the image. Intuitively, each time step contains a series of user interactions. The network then processes this data, learning from the sequence of features across time steps for each sample. For practical reasons, to not lose the possibility to use pre-trained encoders, we introduced LSTM layers only in the decoder, which limits the effect on performance. However, the proposed framework does not prohibit other configurations.

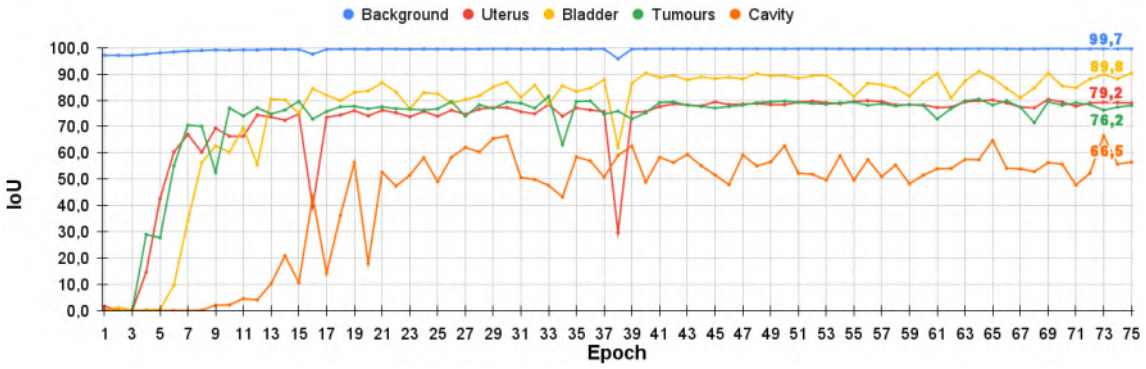


Figure 4.6: Performance on the validation set. The model at the 73rd epoch was chosen for the evaluation.

The encoder was pre-trained on ImageNet (Deng et al., 2009) as a source dataset and subsequently fine-tuned on the proposed FPMRId without frozen layers. While the domain gap is present, transfer learning from ImageNet still proved beneficial for the stability of the training

process and the final model's performance. To counter the dataset imbalance, we use the focal loss (Lin et al., 2017) and dataset-wide precalculated per-class weights.

Data Augmentation and Split. At the time of conducting these experiments, the FPMRId dataset comprised fewer than 113 annotated and validated MRI series (see figure 3.1), specifically 97 series in total. Consequently, all 97 series were utilised for these experiments. Specifically, to avoid inter-slice and inter-patient bias, we denote a single MRI series as the smallest, indivisible element of the dataset and split the dataset as follows: the training set with 77 series containing 2449 slices, the validation set with 10 series containing 308 slices and the test set with 10 series containing 309 slices. Each series originates from a unique patient. Lower-resolution images were padded to 512 by 512, the maximum resolution of a single image in FPMRId at the time these experiments were conducted. We preprocessed all data via normalisation, standardisation and N4BFC (Tustison et al., 2010b), and performed random data augmentation: vertical and horizontal flipping, intensity shifting for brightness, gamma correction for contrast, as well as blurring and unsharp masking for sharpness adjustment.

Training. We trained the network on a single Nvidia P40 GPU with 24 gigabytes of video memory. The chosen batch size was 4. We employed Adam optimizer with standard parameters and a static learning rate of 0.00005. The shape of a single input tensor is the shape of the SIM, which is $C_{\text{sim}} = 1 + 2DN$, where D is the memory's depth and N is the number of classes, including `background`. The network was trained for 75 epochs with the best performance on the validation set achieved at the 73rd epoch. The performance on the validation set given as IoU is shown in figure 4.6. It is shown that the training remains stable with the SIM as an input and the DDG training scheme.

4.5.2 Automated Evaluation

We report an evaluation performed automatically using the virtual user.

Ablation Study. We compared one automatic method and four interactive methods on the created FPMRId, where SDG is Static Data Generation and DDG is Dynamic Data Generation:

1. `Auto`: U-Net with ResNet34 encoder (Le'Clerc Arrastia et al., 2021);
2. `SDG-base`: memory-less system trained with SDG, as described in (Amrehn et al., 2017);
3. `SDG-CIM`: network from `SDG-base` used with a CIM overlay;
4. `DDG-CIM`: system with CIM trained with DDG;
5. `DDG-SIM`: complete proposed system with SIM trained with DDG.

The evaluation setup uses the same network architecture, preprocessing and data augmentation across all systems with a minor network architecture change for `DDG-SIM`. `DDG-SIM` features a ResNet34 encoder with (1-4) a generic decoder or (5) an LSTM-decoder as described in section 4.5.1 and shown in figure 4.5. At test time, clicks are generated via the virtual user.

Comparison with State-of-the-Art. We compared our framework with two classical interactive methods and eight interactive DL methods on the created FPMRId:

Table 4.1: Experimental evaluation results where bold means best and underlined second best. Rows (1-9): existing methods, rows (10-14): ablation study for the proposed framework. `GrabCut`, `VMN`, `NuClick` and `BRS` versions are used per-class, hence `background` metrics are not provided.

Method ↓	**Background**		**Uterus**		**Bladder**		**Tumours**		**Cavity**	
	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice
GrabCut	-	-	17.6	25.1	14.8	21.0	21.7	29.8	8.0	12.4
VMN	-	-	57.6	72.0	78.3	86.1	42.6	55.7	19.8	27.4
NuClick	-	-	23.6	33.1	41.3	54.9	47.0	55.5	52.2	67.7
NoBRS	-	-	36.7	49.7	20.7	30.7	32.4	44.1	7.4	12.1
BRS	-	-	37.4	50.5	21.5	31.6	33.1	44.8	7.8	12.6
RGB-BRS	-	-	37.5	50.6	21.6	31.7	33.1	44.9	7.8	12.6
f-BRS-A	-	-	37.3	50.5	23.9	32.0	33.3	45.1	7.7	12.4
f-BRS-B	-	-	38.3	51.6	23.1	33.3	33.8	45.4	9.6	14.6
f-BRS-C	-	-	37.5	50.7	21.7	31.8	33.0	44.8	7.9	12.7
Auto	99.2	99.6	64.7	78.6	71.9	83.6	60.4	75.3	40.4	57.6
SDG-base	99.1	99.6	61.7	76.3	70.1	82.4	62.5	76.9	21.1	34.9
SDG-CIM	99.3	99.7	66.5	79.9	83.9	91.2	72.8	84.3	29.0	44.9
DDG-CIM	99.6	99.8	77.4	87.3	**87.4**	**93.3**	77.7	87.4	39.6	56.7
DDG-SIM	**99.6**	**99.8**	**79.8**	**88.7**	87.0	93.0	**79.0**	**88.3**	**57.8**	**73.3**

1. `VMN`: volumetric memory network trained with SDG, as described in (Zhou et al., 2022) and inputting extreme clicks;

2. `NuClick`: a segmentation network introduced for microscopy images and trained dynamically in (Koohbanani et al., 2020);

3. `BRS`: a Backpropagating Refinement Scheme (BRS) for mislabeled locations correction, training-less by definition, in (Jang and Kim, 2019);

4. `RGB-BRS`: `BRS` minimised with respect to the RGB image instead of distance maps in (Sofiiuk et al., 2020);

5. `f-BRS variants`: improved `BRS`, `f-BRS` solves an optimization problem with respect to auxiliary variables instead of the network inputs as in `BRS`

   (a) `f-BRS-A`: introduces scale and bias after the backbone

   (b) `f-BRS-B`: introduces scale and bias before the first separable convolutions block in DeepLabV3+ (Chen et al., 2018)

   (c) `f-BRS-C`: introduces scale and bias before the second separable convolutions block in DeepLabV3+ (Chen et al., 2018)

   (d) `NoBRS`: using network architecture from (Jang and Kim, 2019) without `BRS`.

For `VMN` (Zhou et al., 2022), `BRS` variants (Jang and Kim, 2019; Sofiiuk et al., 2020) and `NuClick` (Koohbanani et al., 2020) we use the code and the models made publicly available by the authors and recommended parameters. The models are *resnet34_dh128_sbd* and *NuClick_-Nuclick_40xAll* respectively. We trained `VMN` (Zhou et al., 2022) on our dataset, reducing the batch size to 4 to keep the computation overhead feasible. The metrics are reported in table 4.1.

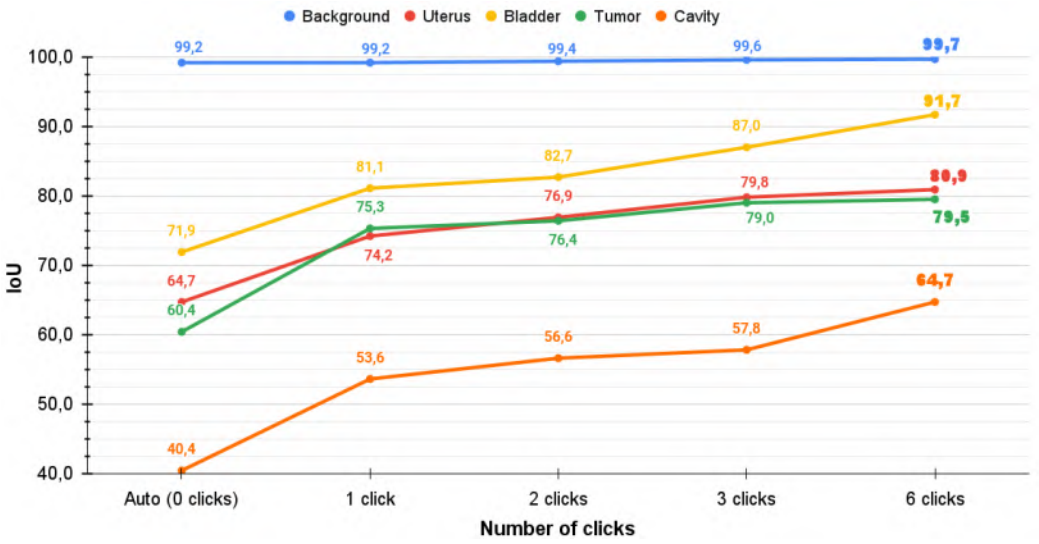


Figure 4.7: DDG-SIM: Influence of the number of clicks simulated at test time on the IoU score, compared with Auto (0 clicks). The strongest improvement presents itself at the first click. Bold means best.

Click Number Influence. An interactive segmentation system refines the segmentation result via user interactions. In essence, this is inputting clicks into the system to provide additional information. Hence, the number of clicks is a key influencing factor in the framework's performance.

We perform a systematic evaluation of the influence of the number of clicks at train and test time on the segmentation accuracy. For this, human user involvement is not feasible due to the number of series and the need to re-segment them for each evaluation setting. Therefore, we perform this evaluation as in section 4.5.2 via the virtual user generating simulated interactions at test time. DDG-SIM is used for the evaluation, where we control only two parameters: the number of clicks simulated at train and at test time. Three setups are provided, each changing click number at training and at testing respectively. They are: (1) training - fixed maximum click number, testing - varying click number; (2) training, testing - equal click number; (3) training - Auto, default DDG-SIM, modified DDG-SIM with rules 2-4 from section 4.4.2 disabled, testing - 0 clicks. The purpose of these setups is as follows: (1) evaluate the influence of the number of clicks at testing on the performance; (2) evaluate the influence of the number of clicks at training on the performance; (3) evaluate the performance of DDG-SIM when no clicks are provided with and without rules 2-4 from section 4.4.2, presence of which should improve the system's ability to automatically segment regions.

For setup (1), the maximum number of clicks simulated at train time is fixed to 3 which is the default value for DDG-SIM, while the number of clicks at test time varies. We then report IoU for all classes when simulating 0 (Auto), 1, 2, 3 and 6 clicks at testing in figure 4.7.

For setup (2) we fix the number of clicks simulated both at train and test time so that they are equal (such as a maximum of 3 clicks at training and exactly 3 clicks at testing) and change them jointly. We then report the IoU for all classes when simulating 0 (Auto), 1, 2, 3 and 6 clicks in figure 4.8.

For setup (3), we evaluate DDG-SIM performance when providing no clicks at testing. We compare Auto, default DDG-SIM and modified DDG-SIM with rules 2-4 from section 4.4.2 disabled. We report IoU for all classes in figure 4.9.

Generalisation Study. We further evaluate the complete proposed system DDG-SIM on two other

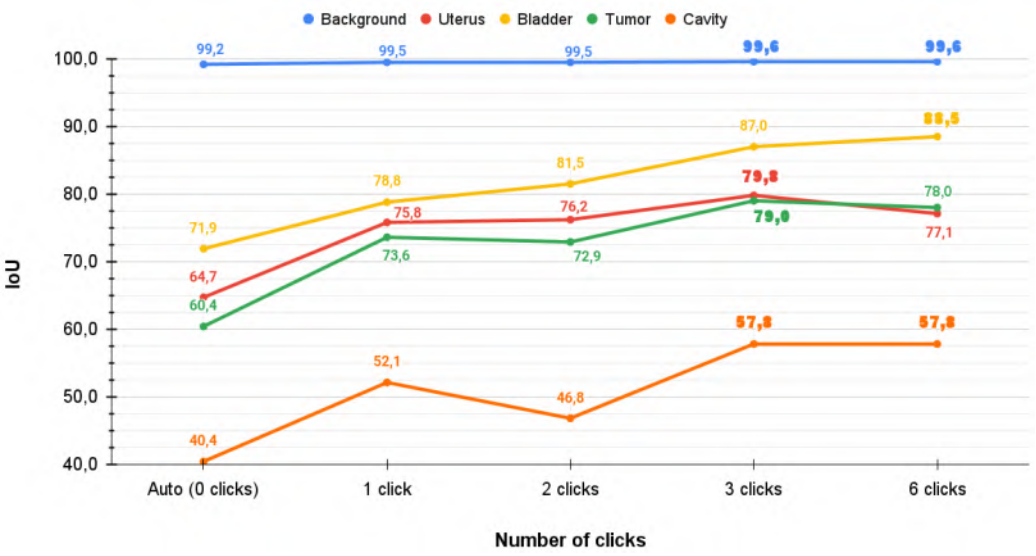


Figure 4.8: DDG-SIM: Influence of the number of clicks simulated at train time on the IoU score, compared with Auto (0 clicks). The number of clicks simulated at training and at testing are equal and change jointly. The overall performance improvement is less noticeable after 3 clicks. Simulating 3 clicks at training is our choice for DDG-SIM with the current data. Bold means best.

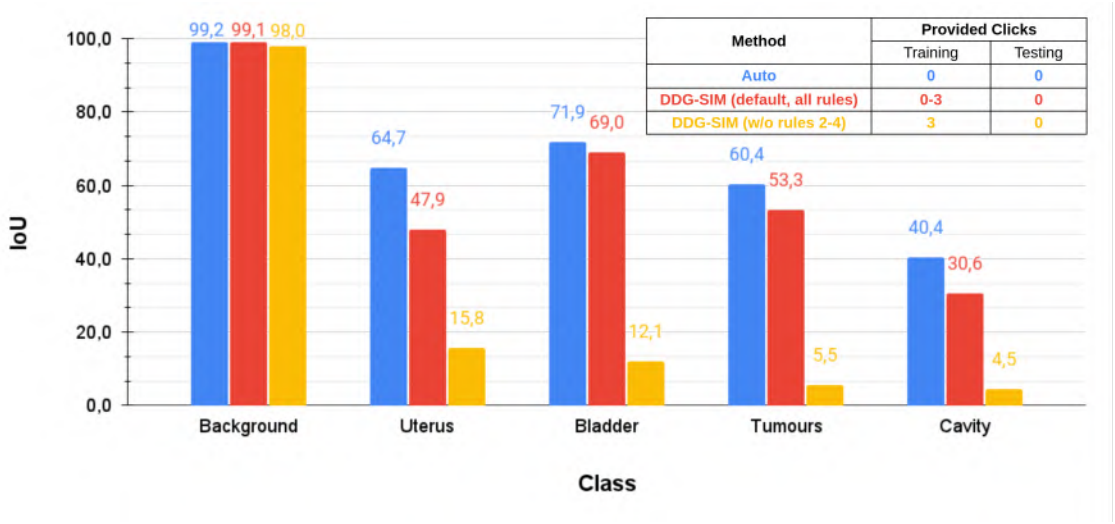


Figure 4.9: DDG-SIM: Performance with 0 clicks provided at testing. Auto, default DDG-SIM and modified DDG-SIM (with rules 2-4 from section 4.4.2 disabled) are compared. Disabling the rules makes automatic segmentation fail. This illustrates the automatic segmentation capability of DDG-SIM as brought by the DDG training process and hence the importance of having varied interaction data when simulating clicks.

tasks with different modality and objects of interest - namely, on liver and pancreas CT segmentation. We use the 'Liver Tumours' and 'Pancreas Tumour' medical segmentation decathlon datasets (Simpson et al., 2019) and compare our framework's performance on these data to the methods participating in the corresponding challenge (Antonelli et al., 2022) as well as VMN (Zhou et al., 2022). Each of the datasets was initially assembled for the task of multi-class segmentation with liver CT targets being liver and cancer, and pancreas CT targets being pancreas and mass (cyst or tumour). While this challenge is aimed at automatic segmentation approaches, a comparison with interactive methods may further prove their feasibility for the tasks usually requiring expert's validation and potential refinement. For VMN (Zhou et al., 2022) we use the code made publicly available by the authors and recommended parameters on these new datasets. We

reduce the batch size to 4 due to the limited GPU availability.

The ground truth labels for the test set were not made available for this challenge. We thus randomly split the publicly available training sets for both `liver` and `pancreas`, using approximately 70%/15%/15% for training, validation and test respectively. As a result, the split is 91/20/20 series for the liver and 198/42/42 series for the pancreas datasets. Effectively, this means that the training is performed on much lower-size datasets than those of the competing methods, which makes it more challenging. To add to this, the key difficulty of these datasets is label imbalance with both large (`liver`, `pancreas`) and small (`mass` or `cancer`) targets. The metrics are reported in figure 4.11 for both `liver` and `pancreas`.

4.5.3 User Evaluation

We performed a user study with `DDG-SIM` involving eleven medical experts, using a specifically developed GUI. All experts have a background in gynaecology, with an exception of two surgeons with specialisation in urology, junior experience level and some experience in gynaecology. For clarity, we assign a letter and a number to each expert as follows: SGS - senior gynaecology surgeon; SR1-2 - senior radiologists; JGS1-3 - junior gynaecology surgeons; JR1-3 - junior radiologists; JUS1-2 - junior urologic surgeons with experience in gynaecology.

We randomly selected 6 test series containing 144 slices in total, where 1 series is used to familiarise the users with the GUI and 5 series are used in a random order for user evaluation. MRI image samples from each of the series can be seen in figure 1.15. We evaluate the user performance in figure 4.10 using mIoU per series for each expert in comparison to the `Auto` method as in section 4.5.2. In the same manner, the elapsed time is compared in figure 4.12. The segmentation results are compared with the `Auto` method in figure 4.4. Figure 4.13 shows mIoU over each class per series.

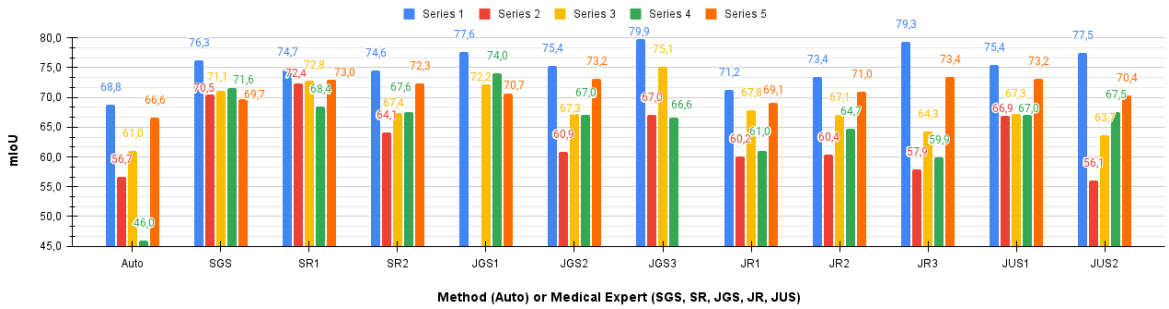


Figure 4.10: User Evaluation: mIoU over all classes per medical expert per series.

4.5.4 Inference Time Analysis

We report the average inference time for a single image and compare it with those of the existing interactive segmentation approaches in table 4.2.

4.6 Discussion

We discuss the results obtained in the previous section.

4.6.1 Automated Evaluation

We discuss results obtained with the virtual user.

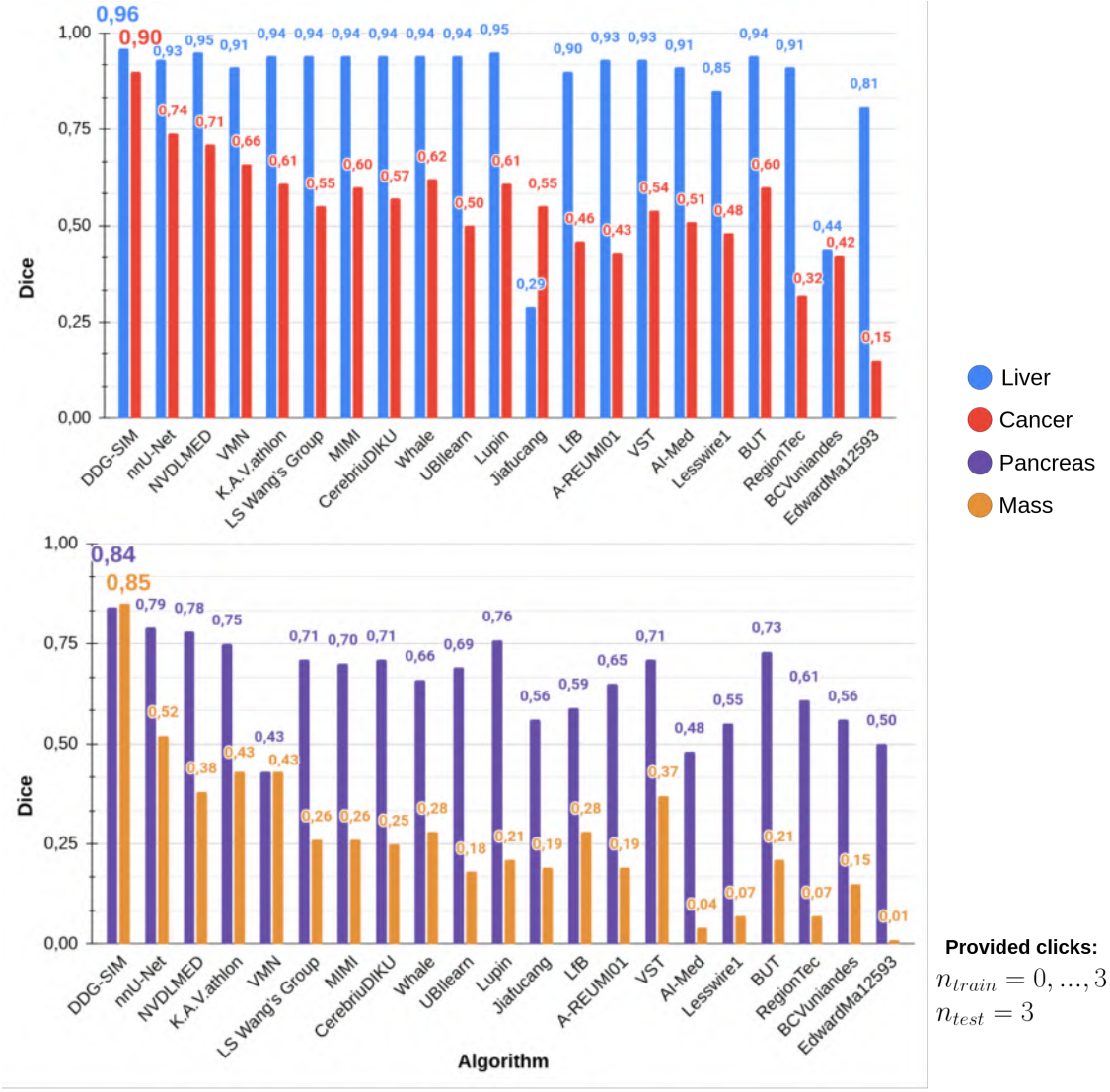


Figure 4.11: `DDG-SIM` experimental evaluation results given as Dice on the medical segmentation decathlon 'Liver Tumours' (`liver` - blue, `cancer` - red) and 'Pancreas Tumour' (`pancreas` - purple, `mass` - orange) datasets in comparison to the automatic segmentation approaches participating in the challenge, where bold means best. `VMN` is a state of the art interactive segmentation approach. The number of simulated clicks is provided for both training and testing in the bottom-right hand corner.

Ablation Study. The metrics are reported in table 4.1, where we observe that the IoU and Dice are in agreement. They show that `DDG-SIM` outperforms, with a substantial margin for `cavity`, a significant margin for `uterus` and `tumour`, a similar result for `background`, and a slight disadvantage for `bladder`, for which `DDG-CIM` slightly outperforms at 87.4% against 87.0% IoU. This demonstrates the robustness of the proposed framework. The ablation study shows a steady increase in performance, starting with `SDG-base` and adding the proposed components towards `DDG-SIM`. `Auto` outperforms both `SDG-base` on `uterus`, `bladder` and `cavity`, and `SDG-CIM` on `cavity`. This can be attributed to SDG, which does not perform well for smaller numbers of interactions. In our experience, the higher the number of interactions at training, the lower the effectiveness of individual interactions at test time. While the opposite is also true, it can be observed from the re-

sults that certain systems may not be able to learn efficiently from a small number of interactions at training. We observe a comparatively lower accuracy for `cavity`, whose IoU lies between 21.1% and 57.8%. We explain this with its low volume, which accounts for only 0.054% of the dataset.

Examining other existing methods, this is also true for `VMN`, which achieves a good performance on `bladder`, but struggles with the more difficult classes. We find that this might be additionally due to the low number of slices in a standard FPMRI scan, where the classes such as `tumour` or `cavity` may be found only on a single slice out of the whole volume in addition to occupying just a few pixels, which may interfere with the approach. Still, `VMN` shows a notable performance on `bladder`, with an IoU of 78.3%, which is competitive but still falls short of `DDG-CIM`'s 87.4%. However, it struggles significantly across the other categories, particularly with `cavity`, where it is greatly outperformed by `DDG-SIM`'s superior IoU of 57.8%.

Interestingly, `NuClick` demonstrates a notably high performance in segmenting the `cavity` class with an IoU of 52.2%. However, it still falls short when compared to `DDG-SIM`, which achieves an IoU of 57.3% for the same class. The relatively high performance of `NuClick` in `cavity` segmentation may be associated with its design and optimization for microscopy image segmentation tasks. The visual characteristics of `cavity` regions in such images may be similar to those that `NuClick` was specifically intended to segment, possibly contributing to its success in this particular class.

The `BRS` and `f-BRS variants` display a range of results, with none matching the `DDG-SIM` scores. Specifically, the `f-BRS-A`, `f-BRS-B`, and `f-BRS-C` methods fall short, with the highest IoU among them for `cavity` being only 9.6%, indicating a substantial gap when compared to `DDG-SIM`. Overall, the superiority of `DDG-SIM` proves it to be a solid segmentation framework in view of the state of the art.

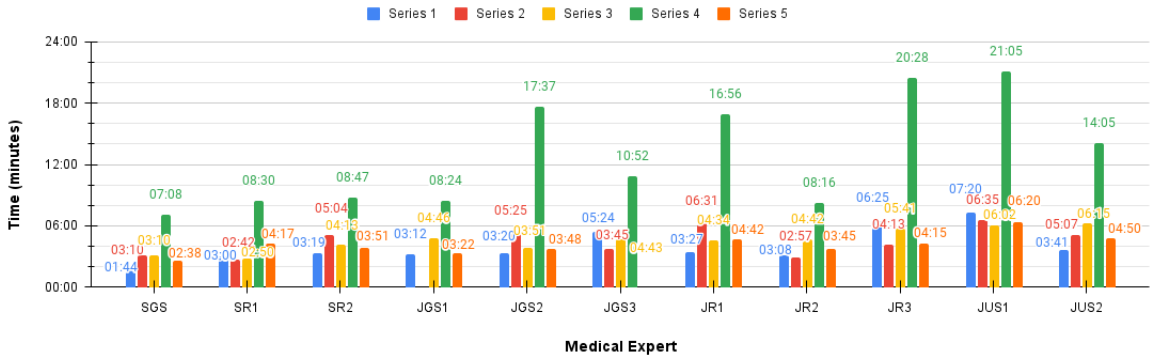


Figure 4.12: User Evaluation: Segmentation time per medical expert per series in minutes.

Click Number Influence. Three setups are provided: (1) training - fixed maximum click number, testing - varying click number; (2) training, testing - equal click number; (3) training - `Auto`, default `DDG-SIM`, modified `DDG-SIM` with rules 2-4 from section 4.4.2 disabled, testing - 0 clicks.

Setup (1). We report IoU for all classes when simulating 0 (`Auto`), 1, 2, 3 and 6 clicks at testing in figure 4.7. The metrics show a substantial improvement of the segmentation accuracy against `Auto` when at least 1 click is provided. Furthermore, a notable growth of accuracy is also observed for all classes when transitioning to 2 clicks. At the same time, while there is a further regular improvement for `cavity` and `bladder` beyond 2 clicks, the other classes improve only slightly. This

can be explained by two factors. First, a single provided click produces an IoU score close to the upper performance boundary achieved by the proposed framework, as seen in the ablation study. This does not leave much room for improvement with a given training set size (77 series, 2449 slices). Second, during training, clicks are currently simulated with a maximum of 3 for all systems. This is done to minimise the amount of interaction required from a human user at test time. The proposed DDG scheme brings the average number of simulated clicks at training even lower, which contributes to the performance stabilising below the maximum click threshold. While still limited by the current performance ceiling, increasing the maximum number of simulated clicks to 6 per class during training may allow to achieve a more stable performance growth with each added click at testing time. At the same time, with 6 clicks a human user evaluation experience would be negatively affected. Indeed, each individual click would bring less improvement, generally requiring more clicks for the same task, which is undesirable in a clinical setting.

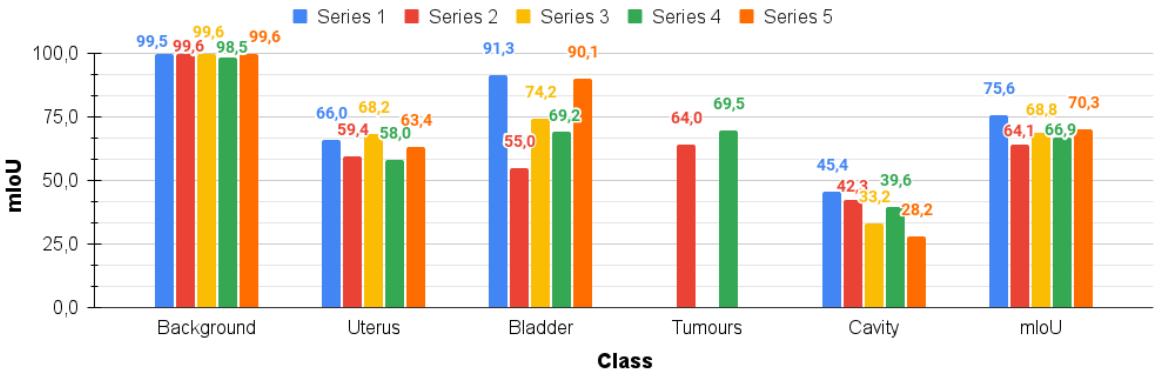


Figure 4.13: User Evaluation: mIoU over all medical experts per class per series.

Setup (2). We report the IoU for all classes when simulating 0 (`Auto`), 1, 2, 3 and 6 clicks in figure 4.8. The metrics show a substantial improvement of the segmentation accuracy against `Auto`, demonstrating robustness of `DDG-SIM` for any number of clicks at training. The strongest performance improvements are observed between `Auto` and training `DDG-SIM` with 1 click, as well as between training `DDG-SIM` with 1 click and with 3 clicks. Performance with 2 clicks shows an overall improvement over that with 1 click, but `cavity` and `tumour` classes show notable and slight performance decrease respectively. This can be explained by the DDG rules we use described in section 4.4.2, which target the increase of individual click efficiency. Specifically, rules 2-4 are such that with the chosen maximum of 1 click at training, it is often the case that no clicks will be simulated at all for many of the labels. This makes the system more reliant on the underlying image features, which places it closer to `Auto`, but still provides a significant performance improvement due the interactivity. In contrast, the chosen maximum of 3 clicks at training allows for more consistent click simulation, which significantly improves performance. At the same time, the maximum of 2 clicks at training is an in-between case, where the actual simulated click number does not seem to be sufficient for the `cavity` and `tumour` labels, which are represented by multiple components or instances of varying size and clarity. In this case, clicks are simulated only for a small number of these components or instances (such as for 1 out of the 7 tumours in a single slice, or only for a part of cavity), which does not allow for consistent learning from clicks and reduces the performance on these classes.

Setup (3). We report IoU for all classes in figure 4.9. The metrics show that when providing

no interactions, default DDG-SIM significantly outperforms modified DDG-SIM, where the chosen maximum number of clicks was consistently simulated for each class during training. Specifically, default DDG-SIM and modified DDG-SIM are respectively 50% against 9% in terms of the mIoU score (background class excluded). Simply put, this figure shows that the use of DDG allows our framework, when used without user interactions, to obtain performance comparable with state-of-the-art fully-automatic segmentation. It also shows that, should the proposed rules 2-4 of DDG were disabled, the framework would fail to perform any meaningful segmentation without user interactions, indicating strong dependence on the number and exhaustiveness of interactions provided at training time. Clearly, the more interactions are provided at training time, the lesser is the network's capability for automatic segmentation in general and for segmentation of un-clicked components in particular. Intuitively, if interactions are scarce, the network focuses more on image features, resulting in higher automation at testing time. While Auto with 59% mIoU outperforms both default and modified DDG-SIM when no clicks are provided, the interactive approaches accuracy can be improved further with additional clicks as shown in figure 4.8, which is not the case for Auto.

Table 4.2: Reported average inference time and standard deviation for DDG-SIM in comparison to existing interactive segmentation approaches, where bold means best and underlined second best.

Method	**Inference Time (ms)**
BRS (Jang and Kim, 2019)	810
Interactive 3D nnU-Net (Isensee et al., 2018)	500
IteR-MRL (Liao et al., 2020)	470
f-BRS-B (Sofiiuk et al., 2020)	226
FocusCut (Lin et al., 2022)	118
FocalClick B0-S1 (on CPU) (Chen et al., 2022)	100
VMN (Zhou et al., 2022)	53
DDG-SIM (ours)	$\underline{47.2 \pm 6.2}$
(Sakinis et al., 2019)	**40**

Generalisation Study. The metrics are reported in figure 4.11 for both liver and pancreas. They show that the proposed interactive framework outperforms the best automatic methods on all classes, with a substantial margin for liver cancer and pancreas mass - 90% against 74% and 85% against 52% respectively and a slight advantage for liver and pancreas classes - 96% against 95% and 84% against 79% respectively. This shows that the proposed framework is generically applicable to other segmentation tasks and medical data types.

4.6.2 User Evaluation

The elapsed annotation time per series is compared in figure 4.12. We note that the segmentation time is low enough to be clinically feasible, even if the users are barely acquainted with the system. Indeed, the average elapsed time for all series is 6'07", which is largely below the reported average of 25' for existing systems. Series 4 was a complex case with 11 tumours and a heavy deformation of the uterus shape, taking 12'55" on average for our system and more than 40' for existing systems. Furthermore, as seen in figure 1.15, each of the series used for user evaluation is challenging in its own manner. While the proposed framework facilitates the segmentation process, interpretation

of the MRI images by the human user remains a task in itself. This explains the elapsed time and the mIoU score discrepancies between senior and junior experts, especially noticeable for series 4 with the peak of 20'28" for junior experts, 8'47" for senior experts in gynaecology and 21'05" for JUS1, the junior expert in urology. Figure 4.10 shows that our framework substantially outperforms automatic segmentation on all data with a lesser improvement for series 5. This is especially noticeable for the difficult series 4, which achieved a score of 46,0% for `Auto` against the average of 66,8% over all interactive human-guided segmentations. Since JUS1 and JUS2 primarily specialize in urology, JUS1 tends to have longer segmentation times, while JUS2 demonstrates reduced accuracy, particularly for series 2. This is attributed to an increased difficulty in image interpretation. Still, the segmentation accuracy of expert JUS1 is on par with the other experts.

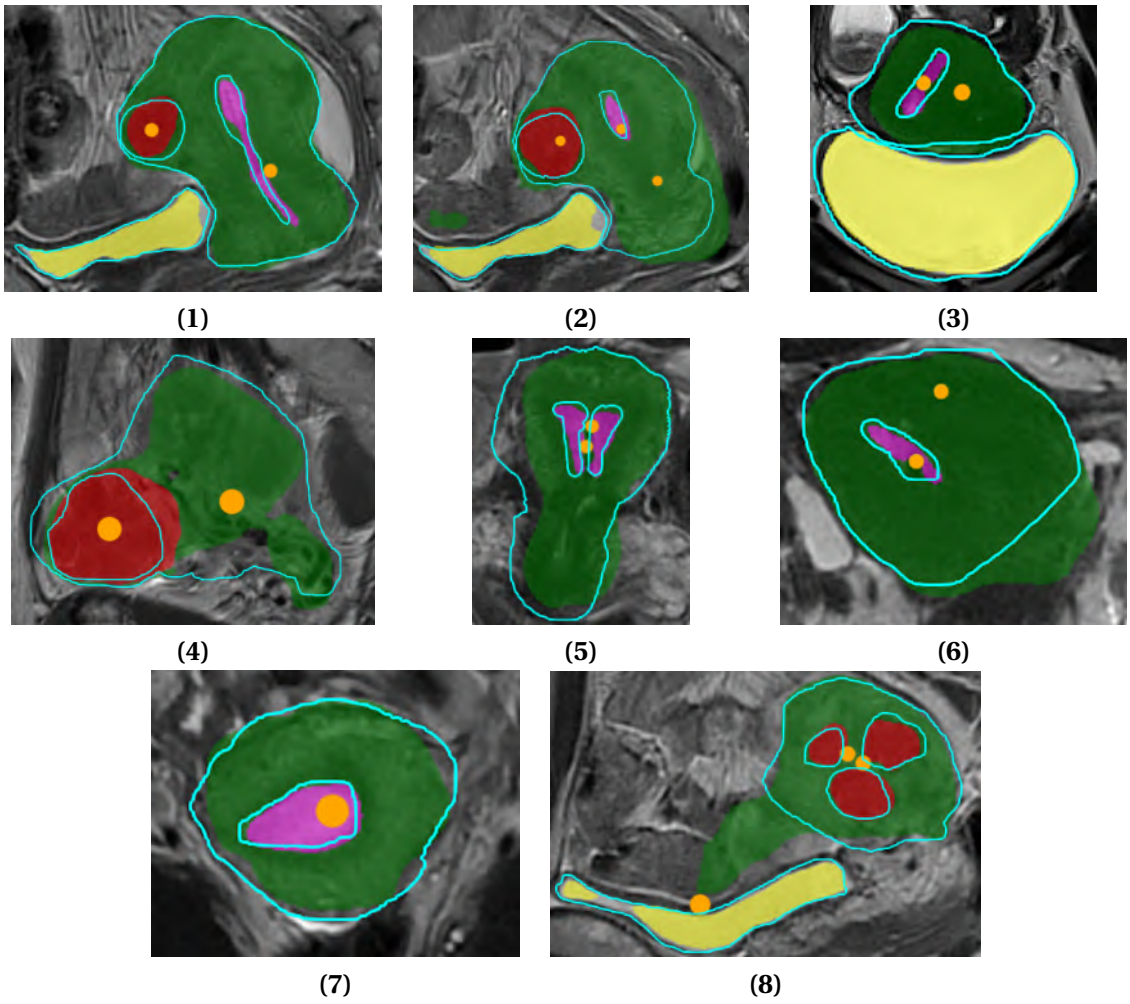


Figure 4.14: Segmentation failure cases, where `uterus` - green, `bladder` - yellow, `tumour` - red, `cavity` - pink, user clicks - orange and ground truth - cyan: **(1,7)** most widespread case with contours having a slight divergence with the ground truth; **(2-5,7-8)** under-segmentation; **(2,4-5,6-8)** over-segmentation. The maximum number of clicks is fixed to three. Additional clicks allow to notably reduce under- and over-segmentation, resulting in segmentations comparable to **(1,7)**. The metrics for `cavity`, present in **(1-3,5-7)**, are affected most strongly in all cases due to its size.

Figure 4.13 shows mIoU over each class per series. We observe a comparatively low accuracy of `cavity` segmentation during user evaluation, similarly to the automated tests. This is because of the small size of the cavity and its lack of clear outer contours. In addition, the slices may split the cavity in a manner that makes it appear in several isolated small components, in which case some

components may be ignored by the users. This creates a variability in the dataset and a potentially large discrepancy between the ground truth and the user segmentation. While this is typical for other FPMRI objects of interest, the cavity's small size strongly amplifies any slight segmentation discrepancy.

4.6.3 Inference Time Analysis

As seen in table 4.2, our framework is on par or significantly faster than existing methods with $47.2 \pm 6.2 ms$ per image. This amounts to approximately 21 FPS, which is well adapted for an interactive clinical application. Hence, usage of SIM with here up to 5 classes does not introduce significant overhead and leaves sufficient room for additional computational complexity (e.g. additional classes or a deeper network).

4.6.4 Implications and Limitations

On the most general level, we find that temporal information associated with user interactions is overlooked in existing methods. Simply put, CIM, which is used in most previous works, does not convey the sequential nature of interactions, discarding the temporal component naturally present in the way the user interacts with the annotation software. However, the proposed SIM conveys this information and its use improves segmentation performance. Furthermore, DDG during training has a significant impact not only on the method's performance, but also on the user experience. Specifically, the ensemble of interaction generation rules in section 4.4.2 allows the network to produce automatic segmentations comparable to fully-automatic methods without user interactions, as well as to segment most of the objects of interest at once by providing a single click for any one of them. This has a large impact for the time-constrained clinical environment.

We show segmentation failure cases in figure 4.14. From our experiments we observe that additional clicks allow to reduce under- and over-segmentation until a case similar to cases 1 and 7 in figure 4.14 is reached. However, remaining divergence from the ground truth notably negatively affects `cavity` metrics due to `cavity`'s size. One limitation of our method is the impact on training speed. The necessity to populate the sequential memory by doing multiple inferences increases the time to process each image. However, the inference time being 47.2 ± 6.2 ms, the training time remains reasonable, even with multiple additional inferences per image. Another constraint is the sequential memory size - increasing memory size C_{sim} increases the computational complexity, especially when using LSTM blocks. However, making the memory too large seems to be counter-intuitive, since the interest lies in having the minimal number of clicks required for a high-quality segmentation at testing, which implies limiting the number of clicks at training and hence C_{sim} in some manner. We show experimentally that, in most cases, providing more than 3 clicks has diminishing returns, and 3 or fewer clicks produce the most significant improvement, suggesting that large C_{sim} is actually counter-productive.

The proposed framework utilises a parameter for the maximum number of clicks, which serves as a starting point for the dynamic data generation described in section 4.4.2. In our experiments we extensively show that 3 or fewer clicks produce the results surpassing those of the comparable frameworks on multiple tasks. However, one can imagine that the maximum number of clicks may change depending on the task, making it a parameter to tune, which might be undesirable if more automation is desired. For this, it might be of interest to select it in an automatic manner for

each image in future work. For example, simply by calculating the segmentation metrics for each click generation at training time (i.e. for each intermediate inference result) and decreasing the probability of adding a new click along with the increase in accuracy.

4.7 Conclusion

We have proposed a general DL-based interactive multi-class image segmentation framework, with a user interaction loop and a sequential interaction memory. The embedded network is trained on dynamically generated data to improve performance and reduce interaction-dependence. We have demonstrated our framework in FPMRI segmentation, using a new dataset. Furthermore, we successfully applied it to the tasks of `liver` and `pancreas` CT segmentation from the medical segmentation decathlon challenge, showing the best overall performance. We have evaluated our framework against existing work in an ablation study with the standard metrics, observed the influence of the number of interactions at test time on performance and conducted a user evaluation, involving 11 medical experts with gynaecology background and varying experience levels to use our software via a specifically-developed GUI. This shows that our framework largely outperforms existing systems in accuracy and drastically reduces the average user segmentation time from 25' to 6'07" when used by either senior or junior expertRegulas.

Chapter 5

Concurrent Data-efficient Annotation and Model Training

5.1 Introduction

ML has gained widespread applicability in recent years, achieving good performance in various fields (Alzubaidi et al., 2021). Still, the performance of supervised ML inherently depends on the size and composition of the annotated training dataset. However, data annotation is expensive, which is especially pronounced for medical images (Castiglioni et al., 2021), which require extensive domain knowledge and commonly present interpretation difficulties. This does not scale well with the growth of the dataset size: considerable expert manpower and amount of time are required when annotating large quantities of data (Castiglioni et al., 2021; Tan et al., 2018). For these reasons, the available data in many ML tasks is imbalanced: non-annotated data greatly exceeds annotated data in quantity.

The research community has made considerable efforts to alleviate the annotation problem and has proposed a wide range of approaches. These can be broadly split into two groups: (1) high-performance annotation predictors (Kirillov et al., 2023; Wasserthal et al., 2023; D'Antonoli et al., 2024), which allow for fast production of annotations; and (2) data-efficient approaches (Adadi, 2021), which reduce the amount of annotated data required for training or lessen the precision requirements for the annotations. The former are represented by predictors trained on sufficient to large quantities of data with foundation models (Bommasani et al., 2021) at the furthest end of the spectrum. The latter may be split in three categories: non-supervised (including semi-, weakly-, self-, unsupervised learning and self-training), knowledge sharing (including zero- and one-shot learning, transfer and multi-task learning) and data augmentation methods (Adadi, 2021; Zhuang et al., 2019; Wang et al., 2019c; Schmarje et al., 2020; Amini et al., 2022). The interactive segmentation solution presented in section 4 falls into the former category. Both high-performance annotation predictors and data-efficient approaches limit or remove the need for human annotation. However, their applicability is limited. High-performing annotation predictors require large quantities of data, which invalidates the purpose of cheaper data annotation. Specifically, while the predictor proposed in section 4 is efficient in minimizing the medical expert's annotation time and performs well on challenging FPMRI, as well as other types of medical data, it does not tackle the initial challenge of obtaining annotations required for its training. Simply, it is implied that annotated data is already available, which is often not the case. In turn, data-efficient approaches

do not make this assumption, but often require complex algorithms and fine-tuning, and may result in less accurate and interpretable models than with supervised learning, which generally offers a simpler way of achieving higher performance when sufficient annotated data is available (Alzubaidi et al., 2021). Thus, finding an efficient data annotation solution to expedite the annotation process from limited available annotated data and abundant non-annotated data is highly required.

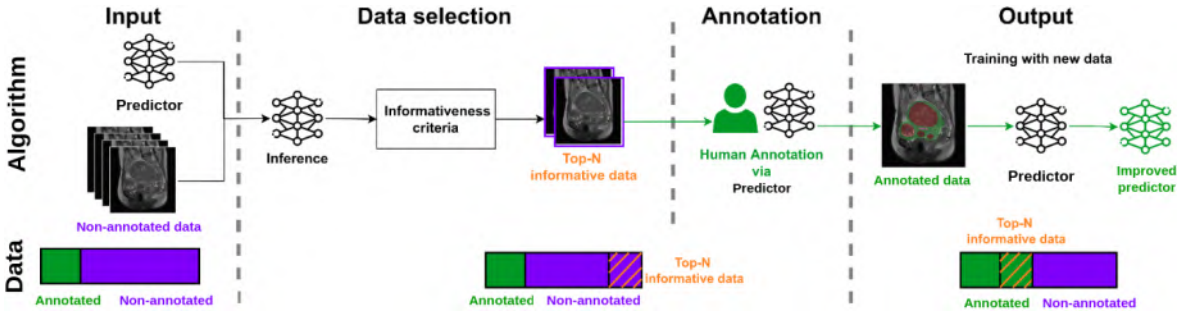


Figure 5.1: Detailed SAIM schematic: a single iteration is shown, only the input changes between iterations.

A general efficient data annotation solution has three goals, which are (1) to produce a high-performing annotation predictor, which (2) effectively uses non-annotated data to (3) output accurate annotations. In other words, the core challenge of the limited annotated data regime lies in how to best utilise non-annotated data to produce an efficient annotation predictor with limited or no queries to the human expert. We find that these goals align with those of SSL (Yang et al., 2023b) in general and with those of its ST (Amini et al., 2022) sub-domain in particular. Specifically, SSL is a type of ML that utilises both annotated and non-annotated data to improve learning performance by assigning pseudo-annotations to non-annotated data. In particular, ST targets generating pseudo-annotations for non-annotated data and using these pseudo-annotated instances in training (Amini et al., 2022). However, while SSL provides a cheap way to obtain annotations, it has a number of drawbacks. First, it lacks theoretical guarantees of performance due to limited theoretical understanding (Yang et al., 2023b; Amini et al., 2022; Ben-David et al., 2008). More precisely, SSL relies on a necessary hypothesis that annotated and not-annotated data have similar distributions (Yang et al., 2023b), which does not always hold in practice, especially in medical imaging (Pulido et al., 2020). Furthermore, it can be shown that even for cases for which the data comply with this hypothesis, the prediction performance might still be poor (Ben-David et al., 2008). Second, SSL is sensitive to error propagation. Incorrect pseudo-labels can degrade predictor performance by introducing noise and errors into the training data. This is especially problematic if the predictor is overconfident in its incorrect predictions. In contrast to supervised learning, these drawbacks make SSL less suitable for tasks where precision is of utmost importance, such as in medical applications, which require guarantees of performance and consistency provided by strong annotations produced and verified by experts.

In contrast, we propose an efficient data annotation framework which allows one to produce strong annotations in the limited annotated data regime. Our framework turns a semi-supervised problem into a supervised one: it produces strong annotations as available in supervised training, instead of pseudo annotations as in semi-supervised training. This is thanks to the interactive predictor used at the heart of our framework, optionally operated by a human expert in order to

validate or iteratively refine produced pseudo-annotations to upgrade them to strong ones.

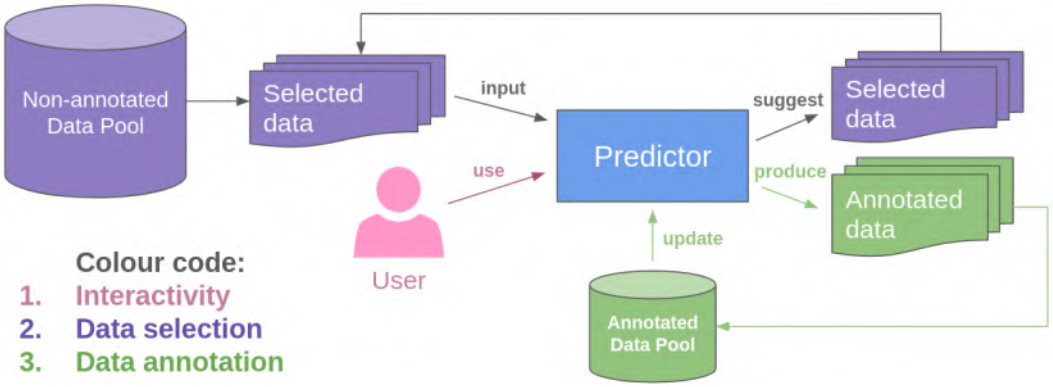


Figure 5.2: A functional schematic of SAIM with respect to the shared predictor.

The base supervised ML paradigm consists of three main steps: (A) data annotation, (B) target predictor training, and (C) evaluation. The system may then loop back to step (A). In the simplest systems, step A may be manual with the use of classical tools such as thresholding (Otsu, 1979), intelligent scissors (Mortensen and Barrett, 1995) and interpolation (Albu et al., 2008), which requires a lot of effort and time. In more advanced systems, this may be improved using sample selection by AL (Zhan et al., 2022; Ren et al., 2020). However, the SOTA shows that replacing the classical tools with a suitable neural annotation predictor boosts annotation performance (Wu et al., 2021). The annotation predictor suggests an annotation that the expert can validate or correct. This raises the question of training this annotation predictor, which existing systems do once a sufficient amount of data has been annotated by classical tools. This is suboptimal for two reasons: (1) annotation is expensive, hence availability of annotated data is very limited in many tasks, especially in medical imaging (Tajbakhsh et al., 2020); (2) neither the annotation predictor nor the classical tools improve as more data is annotated. The main challenge is thus to exploit the data as they are annotated towards training the target predictor, to improve the annotation mechanism itself, including the annotation predictor, which is yet an unresolved problem (Budd et al., 2019; Zhan et al., 2022). The proposed framework addresses these two problems: (1) it requires classically annotated data only once during the interactive predictor pre-training phase, with the number of images needed being as low as several hundreds; (2) the interactive predictor at the heart of our framework is shared between the steps of (A, B and C), making it an evolving annotation tool.

We propose a general framework called SAIM for efficient data annotation at scale, which integrates the three steps of data selection, annotation and training into a single architecture. This is made possible by three key properties of SAIM, which contrast with existing work. 1) deep interactive predictor - the annotation mechanism is based on an interactive neural predictor; hence the predictor can be pre-trained with limited data and still produce quality annotations thanks to the user input. 2) model-sharing - SAIM uses a single model shared between the three steps (A,B and C); the roles of the target predictor and the annotation predictor are thus performed by a single predictor. 3) active data selection - AL is used to maximise the impact of each annotation on the predictor performance, exploiting the current predictor to optimally select data, making the model rapidly improve. To realise SAIM, two key components are required: an interactive neural predictor, which suggests annotations and enables interactive corrections, and a limited quantity

of annotated data used for pre-training and testing. SAIM works in three steps. First, the predictor pre-trained on very limited initial annotations is used for data selection from the non-annotated data pool via AL. Second, the expert uses the predictor to annotate the selected data. Third, the annotations are added to the training data for predictor update. The system then loops back to the first step and continues until stopped or all available data are annotated. As a result, SAIM allows one to efficiently annotate massive datasets from very limited initial annotations, while keeping the single shared predictor up-to-date and deployable.

Table 5.1: Key feature differences between SAIM base and advanced versions, which we denote `SAIM-base` and `SAIM-advanced` respectively.

Component	**Version**	
	`SAIM-base`	`SAIM-advanced`
Data selection	Entropy-based	Loss-prediction-based, Class weighting
Data annotation	Interactive predictor (section 4)	
Predictor update	Re-training	Fine-tuning

Our main contribution is SAIM, which is the first general ML framework to integrate data selection, annotation and training into a single architecture by model-sharing. Two key proposed ideas required to realise such an integrated framework are the use of deep interactive annotation with the shared model and the use of AL for efficient data selection with said shared model.

We evaluate SAIM and compare it to existing systems in emulated annotation scenarios in an automated manner with fully-annotated segmentation datasets on five tasks. First, on multi-class semantic MRI segmentation of the female pelvis on FPMRId. Second and third, on multi-class semantic liver and pancreas CT segmentation, using the 'Liver Tumours' and 'Pancreas Tumour' public medical segmentation decathlon datasets (Antonelli et al., 2022). Fourth, on cardiac MRI segmentation using the public ACDC dataset (Bernard et al., 2018), on which we validate SAIM against the state-of-the-art SSL approach UniMatch (Yang et al., 2023a). Fifth, on natural image segmentation using the Pascal VOC 2012 (Everingham et al., 2015) public dataset expanded with annotations from the SBD dataset (Hariharan et al., 2011), on which we validate SAIM against the state-of-the-art ST approach ST++ (Yang et al., 2022). To assess the impact of the individual components that make up the SAIM framework, we introduce and compare two versions: base and advanced, which are outlined in table 5.1. These versions differ based on the underlying methods for the two key SAIM components: the predictor update mechanism and the data selection process. The latter is further augmented by the addition of class-based weighting. Specifically, the advanced version is designed to enhance the architecture in three ways: (1) reduce the predictor update time, (2) increase the impact of each annotated image on the predictor performance and (3) reduce the class imbalance effects. We conduct a comprehensive ablation study of the SAIM framework architecture, evaluating the impact of each new method introduced in the advanced version compared to the base version. We evaluate SAIM against the two most relevant domains - SSL and ST domains, represented by state-of-ther-art approaches on the Pascal VOC 2012 and ACDC public datasets respectively. This means a direct comparison with five SSL approaches on

ACDC, as well as four ST and three SSL approaches on Pascal VOC. We demonstrate SAIM in a real annotation scenario of kidney MRI segmentation from the AMOS dataset (Ji et al., 2022) with a human user and a 1 to 30 annotated to non-annotated data ratio. We estimate the time gain as compared to 3D Slicer, where SAIM allowed to double the total number of AMOS kidney MRI annotations in 2.3 hours against 10.0 hours for 3D Slicer using classical tools. SAIM jumpstarts efficient interactive annotation from limited annotated data and minimises the amount of data to annotate, while iteratively improving performance.

5.2 Related Work

The main goal of this work is efficient data annotation. Specifically, SAIM modifies the step (A) of the base ML paradigm discussed in section 5.1, where step (A) is data annotation, step (B) is target predictor training, and step (C) is evaluation. Step (B) is typically achieved by involving an expert engineer or by continual learning (Hadsell et al., 2020), but the method of training is not in the scope of our contributions.

5.2.1 Categorising approaches

Data annotation is a broad subject approached from multiple angles in the literature. However, there is no established categorisation of existing data annotation approaches (Heim et al., 2018; Langlotz et al., 2019; Willemink et al., 2020a; Bhagat and Choudhary, 2018). We propose to categorise these approaches based on which of the two core aspects of data annotation they address. The annotation mechanism approaches focus on the mechanism used to annotate the data, ranging from classical tools to high-performance annotation predictors. The data efficiency approaches focus on how to most efficiently utilise the data. Approaches in the annotation mechanism category can be further categorized, depending on whether the data annotation mechanism is kept fixed or is improved as annotation progresses, leading to two groups of approaches we call static and dynamic respectively.

5.2.2 Approaches in the annotation mechanism category

The data annotation mechanism may use classical tools, neural predictors, or a combination of both. The classical tools are non-neural and non-trainable, hence not specific to a single task or domain, allowing for a wide applicability. They may strongly vary in functionality and complexity: for example, in image annotation, we find the intelligent scissors (Mortensen and Barrett, 1995), GrabCut (Rother et al., 2004) and Random Walker (Grady, 2006b). In contrast, the neural predictors have to be trained or be already available pre-trained, and are generally specific to a task and a domain. They can be fixed (i.e. static) or improved by training as data annotation proceeds (i.e. dynamic). This implies two general statements: (1) an approach which uses exclusively classical tools is necessarily a static approach, and (2) a dynamic approach necessarily uses a neural predictor, which improves with time.

Fixed neural predictors require huge initial training datasets such as MS COCO (Lin et al., 2014b), AbdomenAtlas-8K (Li et al., 2024), Open Images (Benenson and Ferrari, 2022), as shown in a survey of over 100 segmentation predictors (Minaee et al., 2021). However, the availability of such datasets is limited for many tasks, especially in medical image analysis (Tajbakhsh et al.,

2020). Dynamic predictors are generally used in continual learning (Hadsell et al., 2020) and in ST (Amini et al., 2022). The main focus of continual learning is the mechanism of incorporation of new data into an existing predictor and not data efficiency. This is not the case for ST, where a dynamic predictor is used in a SSL manner, adding predicted pseudo-labels to the dataset with the goal of improving the predictor. The annotation cost is thus reduced. However, ST approaches either use classical tools to refine the pseudo-labels into strong labels to permit supervised training, which is inefficient, or otherwise suffer from error propagation (Amini et al., 2022).

5.2.3 Approaches in the data efficiency category

The data aspect is generally addressed by optimising the way the available data is used. We find SSL in general, and ST specifically the closest domains to SAIM due to the initial data setup: only a fraction of data is annotated, while the rest has no annotations at all. However, the key difference of SAIM is that the predictor is trained in a supervised manner with strong annotations, made possible thanks to model-sharing between the task, which is not the case for existing approaches in both SSL (Yang et al., 2023b) and ST (Amini et al., 2022). Furthermore, any approach may be complemented by data selection via AL (Zhan et al., 2022; Budd et al., 2019; Ren et al., 2020), which means selecting the most informative data to annotate, instead of annotating all available data indiscriminately. SAIM incorporates AL to further reduce the annotation cost via either classical entropy-based data selection (Zhan et al., 2022) in the base version or advanced state-of-the-art data selection according to the predicted loss, inspired from (Yoo and Kweon, 2019).

5.2.4 Approaches in both categories

There is a synergistic effect in addressing both the annotation mechanism aspect and the data aspect in the same solution, as shown in the ST approach (Tajbakhsh et al., 2020), which is where SAIM belongs. On a basic level, the components of such a synergistic solution would be the following: (1) a ready high-performance annotation predictor, (2) as little as possible data to annotate, and (3) a way to exploit this limited data for the target task in a supervised manner as if a large quantity of data were used. In contrast to (Tajbakhsh et al., 2020), with SAIM, we show that (1) does not require an already available high-performance predictor trained on a large dataset and instead a minimally pre-trained interactive predictor. Further, we show that this single neural predictor can be shared between (1) and (3) and is sufficient, instead of two predictors - one for data annotation and another one for the target task.

5.2.5 Existing systems

Most of the existing systems such as Synapse 3D (Fujifilm, 2024), 3D Slicer (Kikinis et al., 2013), MITK (Goch et al., 2017), Supervise.ly (Supervisely OU, 2024) and others (Aljabri et al., 2022) implement static approaches. They provide access freely or commercially to a large variety of classical tools and fixed neural predictors. For example, (Liu et al., 2019; Yu et al., 2015) are static approaches based on classical tools. They use AL as a data selection policy in reinforcement learning (Liu et al., 2019) or train a neural classifier to perform data selection (Yu et al., 2015).

SAIM shares its goal with SSL. As in SSL, SAIM uses non-annotated and annotated data jointly to improve learning performance. More precisely, SAIM is inspired by ST and one of its building blocks - dynamic predictor, which is used to produce the annotations for the non-annotated data

pool. Typically, among others, approaches in SSL and ST are compared by the following metrics: (1) time gain over annotating more data classically and (2) performance gain when annotating more data classically, targeting smaller performance difference, which is the case for SAIM as well. For these reasons, we find the state-of-the-art SSL and ST approaches (Yang et al., 2023a) and (Yang et al., 2022) close to ours. (Yang et al., 2023a) is static system based on FixMatch (Sohn et al., 2020), adopting a consistency regularization framework with a student-teacher approach by proposing two perturbation streams. While substantially different from SAIM, which does not adopt the consistency regularization framework, (Yang et al., 2023a) do extensive experiments for metric (2) and serve as a strong competitive baseline. In turn, (Yang et al., 2022) is a dynamic system, which uses multiple predictors' stability-based consensus to choose the most reliable annotations in order to re-train the main predictor, which means that produced annotations are weak. In contrast, SAIM produces strong annotations and uses a dynamic interactive predictor for both annotation and data selection.

5.2.6 Closest works to SAIM

The two closest works to SAIM are the non-medical object detection annotation system (Wong et al., 2019) and the MONAI Label toolbox (Diaz-Pinto et al., 2022). System (Wong et al., 2019) is a dynamic system and is a typical example of a large-scale or crowd-sourced annotation approach. Images selected based on the Euclidean distance are segmented by a pre-trained neural predictor. Annotation corrections are then done with classical tools by real users. The neural predictor is periodically re-trained from the corrected annotations. The MONAI Label toolbox (Diaz-Pinto et al., 2022) is a static system. It combines classical tools provided by 3D Slicer and two fixed neural predictors, an automatic one and an interactive one. In contrast, SAIM is a dynamic system, which (Diaz-Pinto et al., 2022) is not, uses a dynamic interactive neural predictor, which (Wong et al., 2019) does not, and uses the predictor to perform data selection, which neither of (Wong et al., 2019; Diaz-Pinto et al., 2022) do. This unique combination is the key to enable model-sharing, where the single predictor is used for data selection, interactive data annotation and undergoes improvement with new data via re-training or fine-tuning, which is not featured in any existing approach and system.

5.3 Methodology

5.3.1 System Overview

On a general level, we build SAIM as shown in the row 'Algorithm' of figure 5.1. As inputs, SAIM requires the following: (1) a minimally pre-trained interactive predictor, (2) a non-annotated data pool, (3) a set of criteria to perform data selection and the number N of samples selected for annotation at each iteration, which is determined as a function of the dataset size and the annotation capacity. We originally developed SAIM for 3D image segmentation of medical scans including MRI and CT, but the framework is generic and not restricted to a specific task, domain, modality or predictor architecture. Indeed, the suitable interactive neural predictor can be obtained from any trainable interactive ML architecture. This means that at inference it should take both the non-annotated data (e.g. an image) and the user corrections as inputs and output an annotation. Some examples of such interactive predictors are (Amrehn et al., 2017; Liao et al., 2020; Mikhailov

et al., 2024) or notably SAM (Kirillov et al., 2023). Such architectures are generally reusable for different tasks, and by design fit into SAIM in a plug-and-play fashion. Specifically, we reuse the neural interactive system (Mikhailov et al., 2024) presented in section 4, which inputs user clicks and outputs a segmentation mask for each class. The neural interactive predictor is a key component of SAIM, which allows it to quickly produce better annotations through interactive refinement as opposed to static approaches.

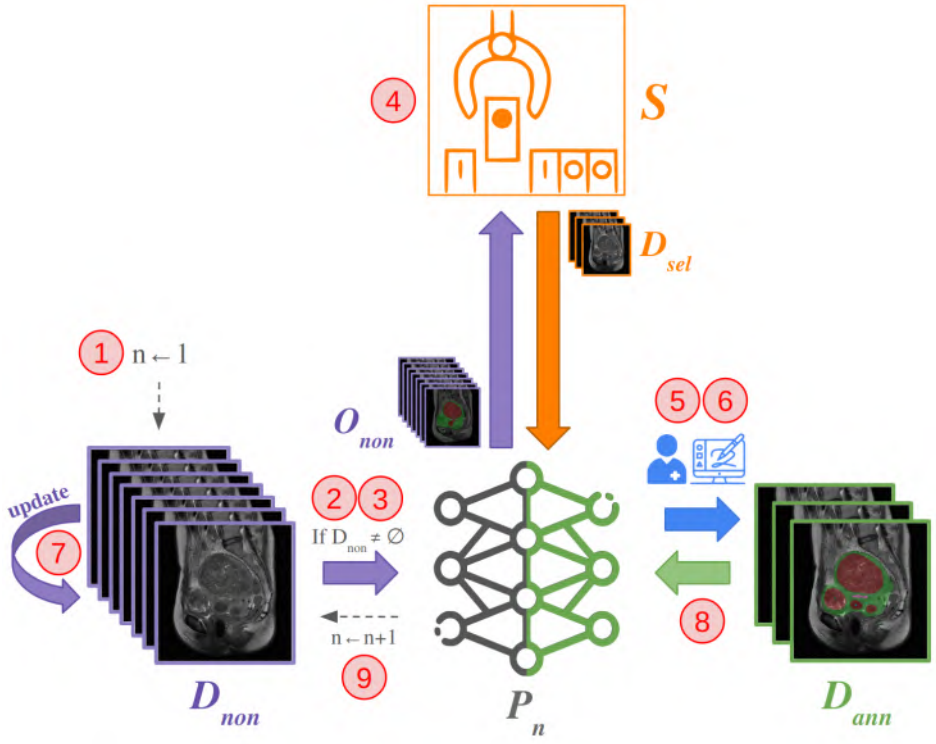


Figure 5.3: Visualization of the SAIM logic as detailed in algorithm 5.1, with corresponding algorithm steps marked by red circled numbers in the image.

SAIM's logic is outlined in algorithm 5.1 and is visualised in figure 5.3 in direct correspondence to the algorithm. SAIM operates in iterations. At each iteration, SAIM inputs a predictor and outputs an updated predictor and new annotated data. Within an iteration, SAIM goes through three inner steps. Model-sharing is implemented by having a single predictor shared between these three steps as shown in figure 5.2. Concretely, the predictor is shared in its entirety and not via parameter preservation as in gradient-based continual learning approaches (Hadsell et al., 2020). The three inner steps are as follows. First, we perform data selection from the non-annotated pool by doing an inference on the non-annotated data and applying predefined selection criteria to the predictor's output. Second, we perform interactive annotation of the selected data using the predictor within the interactive neural annotation mechanism, where it is up to the human user to validate or refine the automatically obtained annotations. Simply, the predictor serves as both an automatic and an interactive annotation tool. The annotated data pool is then expanded with this newly annotated data. Third, we update the predictor by re-training it with the expanded annotated data pool in the base version of SAIM, or fine-tuning it in the advanced version. Iteration $n+1$ therefore improves on iteration n by two factors: (1) the quantity of annotated data increases owing to data selection and data annotation done in iteration n, (2) the predictor at iteration $n+1$ is thus improved by benefiting from the increased quantity of data compared to iteration n. The

iterations continue until SAIM is stopped or all available data are annotated. Thus, at each iteration SAIM uses a shared predictor to select data, annotate the data interactively under control of a human user and employs this newly annotated data in addition to the current annotated training set to update the predictor.

Algorithm 5.1: SAIM pseudo-code

Input: P_n, D_{non}, S % Pre-trained predictor, Non-annotated data pool, Data selection criteria
Output: P_n, D_{ann} % Updated predictor, Annotated data pool
1: $n \leftarrow 1$ % Initialise the iterator
2: **while** $D_{non} \neq \emptyset$ **do**
3: $O_{non} \leftarrow P_n(D_{non})$ % Perform inference to obtain predictions on non-annotated data
4: $D_{sel} \leftarrow S(D_{non}, O_{non})$ % Select the most informative samples
5: $D_{sel} \leftarrow \text{Annotate}(D_{sel}, P_n)$ % Annotate with the predictor and human user
6: $D_{ann} \leftarrow D_{ann} \cup D_{sel}$ % Update the annotated data pool
7: $D_{non} \leftarrow D_{non} \setminus D_{sel}$ % Update the non-annotated data pool
8: $P_{n+1} \leftarrow \text{Train}(P_n, D_{ann})$ % Update the pre-trained model
9: $n \leftarrow n+1$ % Update the iterator
10: **end while**

5.3.2 Data Selection

SAIM starts with an interactive predictor pre-trained on a generally small quantity of annotated data and is immediately ready to produce new annotations. However, during the first iterations, extra user interactions may be required to correct the output of such a predictor, resulting in prolonged annotation time. It is thus crucial to speed up the annotation process and to ensure the performance of the predictor is improved rapidly going forward. We address this point by maximising the impact of each individual annotation. Specifically, we perform data selection as the first step in the SAIM framework. This is done via AL, which involves selecting the most informative data from a pool of non-annotated data and then requesting annotations for this data from a human expert. By doing so, the predictor can achieve high accuracy with fewer annotated examples required for training.

Data selection in AL is often performed based on informativeness. Informativeness refers to the degree of usefulness of the selected data in improving the performance and generalisation of a predictor if added to the training set. Data informativeness is evaluated by informativeness criteria, which allow us to directly select the images benefiting predictor improvement for subsequent annotation. We split existing informativeness criteria into two groups: external or internal. External criteria involve additional information such as image meta-data reflexive of clinical characteristics. Internal criteria are based on the image data itself and may or may not involve the predictor. SAIM is not restricted to neither internal, nor external group, and can be configured to be used with any one or both. Internal criteria are generally based on uncertainty or representativeness (Budd et al., 2019; Zhan et al., 2022). Uncertainty-based methods select samples with high aleatoric or epistemic uncertainty, where the first comes from inherent randomness or noise in the data, and the second comes from lack of knowledge or data. In turn, representativeness-based methods select samples representative of the non-annotated data pool. The key assumption for both uncertainty-based and representativeness-based methods is that the selected samples, once annotated, will substitute the need to annotate all the data.

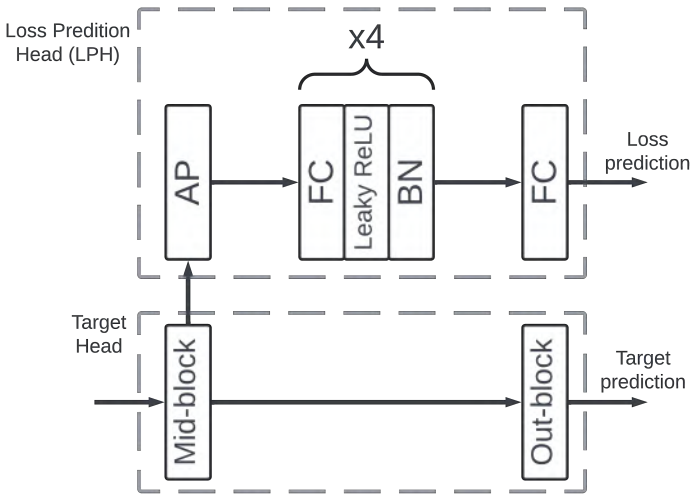


Figure 5.4: Schematic of the architecture of the Loss Prediction Head (LPH) and its connection to the main head, which produces the output for the target task. FC is a fully-connected layer, AP is average pooling and BN is batch normalisation.

We implemented SAIM with two criteria, corresponding to the base and advanced versions. Both criteria are internal, uncertainty-based and dependent on predictor output: (1) in the base version, it is a classical entropy informativeness criterion, and (2) in the advanced version, it is a state-of-the-art predicted loss informativeness criterion.

Entropy informativeness criterion. Entropy is a classical informativeness criterion. A higher entropy is obtained for images with target classes having closer probabilities pixel-wise. Simply put, images with ambiguous predictions overall are considered more informative and are selected for subsequent annotation (Budd et al., 2019; Ren et al., 2020). For classification, entropy is calculated according to the following formula (Shannon, 1948):

$$H = -\sum_{c=1}^{C} p_c \log(p_c), \tag{5.1}$$

where C is the number of classes and p_c is the probability of the sample belonging to the c-th class. For the 3D image segmentation case with a single series we do the following: first, calculate the entropy for each pixel from the predicted class probability distribution; second, in the advanced version, weight the entropy per-class in order to alleviate class imbalance and finally, average the weighted entropy over all pixels. Hence, the formula (5.1) is updated to the following:

$$H_{series} = -\frac{1}{I}\sum_{i=1}^{I}\sum_{c=1}^{C} w_c p_{ic} \log(p_{ic}), \tag{5.2}$$

where I is the total number of pixels and is fixed, p_{ic} is the probability of the i-th pixel belonging to the c-th class and w_c is the weight for class c, with the condition that $\sum_{c=1}^{C} w_c = 1$.

With entropy we perform data selection in three steps: (1) the predictor outputs probability maps for all samples in the non-annotated data pool via inference; (2) the entropy of each sample is evaluated, resulting in a score; (3) the top N scoring samples are selected for annotation, while the rest of the data remains in the non-annotated pool. The schematic of the data selection step is shown in figure 5.1.

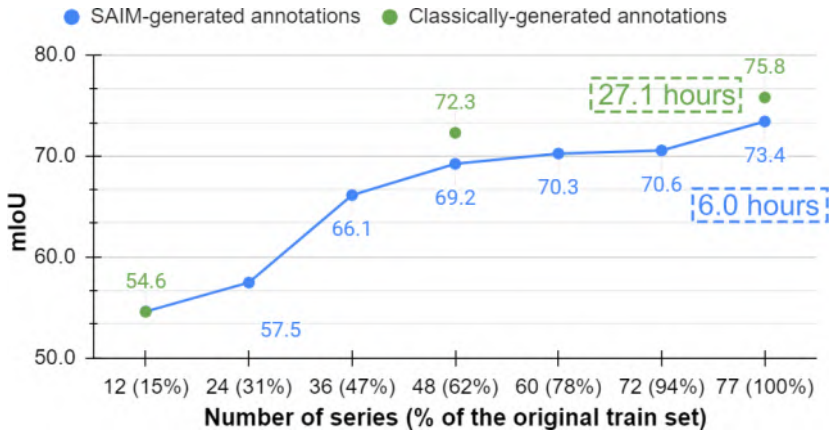


Figure 5.5: `SAIM-base` experimental evaluation results given as mIoU at each iteration on FPMRId, where: green - performance using annotations created in 3D Slicer and MITK using classical tools, blue - performance using SAIM-created annotations. Annotation time is reported excluding pre-training data annotation with classical tools.

Predicted loss informativeness criterion. Loss prediction informativeness criterion is based on predicting the loss for non-annotated samples. Samples with the highest predicted loss are picked for subsequent annotation. This is done using a small additional head we further call LPH attached to the predictor, inspired from (Yoo and Kweon, 2019). Specifically, in (Yoo and Kweon, 2019), the network learns to predict loss by minimizing the Loss Prediction Loss (LPL). This loss is calculated from the difference between the loss predicted by the LPH and the target loss of the target head. The key idea in (Yoo and Kweon, 2019) is to ignore the overall scale of the real loss, which decreases as learning progresses. If this decrease is learned, the LPH will not generalize well. To address this, the LPL is designed to compare pairs of samples, allowing the network to avoid learning the natural decrease in loss and instead fit the exact loss values.

Compared to (Yoo and Kweon, 2019), loss prediction in SAIM is different in three ways. First, we found that the original complex multi-scale architecture does not perform well with time series data as in (Mikhailov et al., 2024, 2023) (see section 4) and replaced it with a leaner Multilayer Perceptron (MLP) architecture branching from the target prediction head's penultimate layer as shown in figure 5.4. Second, we found that (Yoo and Kweon, 2019) does not perform well with lower batch sizes. This might be due to the LPH's inability to learn to fit the exact loss value with a limited number of data pairs. Specifically, as shown in table 5.3, we use a low batch size of 10, constrained by the interactive predictor, which inputs the SIM along each image. Simply, the SIM is a set of masks representing the sequence of user interactions, which occupies additional memory and reduces the batch size, as explained in detail in section 4. To address the issue of a smaller batch size, we propose forming pairs not between samples, but instead between two samples' classes, resulting in an increased pair number P_{num} with $P_{\text{num}} \leftarrow P_{\text{num}} \times C$, where C is the number of classes. The LPL formula is provided in equation 5.4. Third, we found that (Yoo and Kweon, 2019) is not designed to predict loss per class and, hence, does not perform well for datasets with pronounced class imbalance, which we addressed by weighting the loss predicted by the introduced MLP similar to the formula (5.2). The final loss function L_{final} for the predictor is thus formulated as:

$$L_{\text{final}} = L_{\text{focal}} + \lambda \cdot L_{\text{lph}}, \tag{5.3}$$

where L_{focal} represents the focal loss for the target prediction head, L_{lph} denotes LPL - the loss for the LPH, and λ is the scaling constant that determines the relative weighting between the two losses. In turn, L_{lph} is defined as:

$$L_{\text{lph}}(\hat{l}_i, \hat{l}_j, l_i, l_j) = \max\left(0, -\mathbb{1}(l_i, l_j) \cdot (\hat{l}_i - \hat{l}_j + \xi)\right), \quad \text{s.t. } \mathbb{1}(l_i, l_j) = \begin{cases} +1, & \text{if } l_i > l_j, \\ -1, & \text{otherwise.} \end{cases} \tag{5.4}$$

where l_i, l_j and $\hat{l}_i, \hat{l}_j$ are target and predicted losses for samples i and j belonging to the same class c. ξ is a pre-defined positive margin.

With loss prediction we perform data selection in two steps: (1) the LPH outputs predicted losses for all samples in the non-annotated data pool via inference, (2) the top N scoring samples are selected for annotation, while the rest of the data remains in the non-annotated pool as normally.

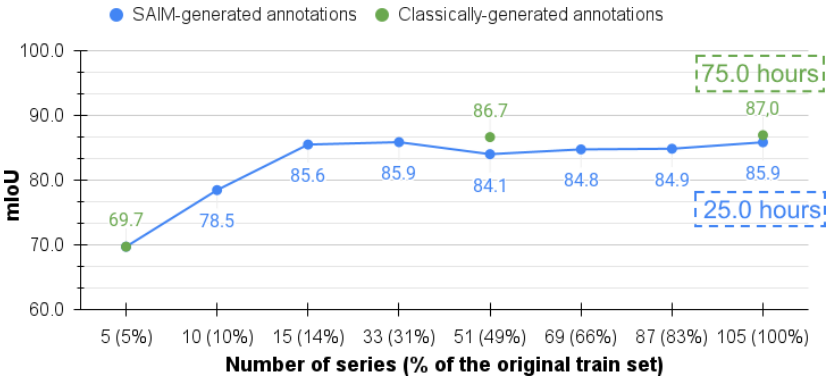


Figure 5.6: `SAIM-base` experimental evaluation results given as mIoU at each iteration on Liver CT dataset, where: green - performance using annotations produced using classical tools, blue - performance using SAIM-created annotations. Annotation time is estimated excluding pre-training data annotation with classical tools.

5.3.3 Data Annotation

Once data selection is performed, the selected samples are passed to the human user for annotation using the shared interactive predictor. Indeed, SAIM requires an interactive predictor, which suggests an annotation to the user, and is capable of accepting the subsequent user corrections. Along with data selection, interactivity is a key property of the SAIM architecture, which ensures that annotations of sufficient quality are produced even if the interactive predictor is initially pre-trained on a limited amount of data. The interactivity allows the predictor to achieve a far better annotation quality than that of an equivalent automatic system by design.

SAIM does not depend on a specific architecture of the interactive predictor, which can be any trainable ML architecture, as long as it accepts user corrections and outputs the annotation. To demonstrate SAIM, we use the interactive system (Mikhailov et al., 2024) presented in section 4, which focuses solely on interactive image segmentation. This system consists of an embedded network, a user interaction loop and an interaction memory. It inputs an image, user interaction masks and optionally a segmentation mask, if available from previous steps, for each class. It

Table 5.3: General training configuration and main parameters, where default denotes standard parameters, target model denotes the body of the network and the target head (the whole network except LPH), and LR denotes learning rate.

Parameter	**Value**
GPU	×1 RTX 4090 24Gb
Batch size	10
Epoch num.	early stopping
Optimizer	adam (default)
Scheduler	ReduceLROnPlateauScheduler
Target model: LR	0.00005
LPH: LR	0.00001
LPH: λ (scaling constant)	0.000003
LPH: ξ (positive margin)	1.0 (default)
Focal Loss gamma	2.0 (default)

outputs the segmentation probability maps. User interaction masks contain user clicks indicating foreground and background for each class. The interaction memory keeps track of the user corrections throughout the interactions by storing a sequence of states, where each state is a pair of user input and corresponding segmentation output. With interaction memory it is shown that using the temporal aspect of user interactions (namely, the user interaction sequences) for training improves performance. This system comes with a specific training approach, where the user corrections at training time are automatically generated on-the-fly from the annotated dataset by means of a virtual user simulating interactions. At test time, this system is used by the human user via a general-purpose GUI. In our experiments, we use the system with a human user to evaluate its impact in a real annotation scenario. We also reintroduce the virtual user at test time to conduct an extensive statistical evaluation in emulated annotation scenarios where the complete dataset annotation is already available, and human user involvement is not feasible due to the sheer volume of data.

Table 5.5: Main libraries, operating system, and versions.

Name	**Version**
PyTorch	2.2.2
MONAI	1.3.2
CUDA Toolkit	12.3
Segmentation Models	0.33
Ubuntu	22.04.3 LTS

5.3.4 Predictor Update

Once the data is annotated, it is used to extend the current training set, after which the predictor is updated, as shown in figure 5.1. The update can be a full re-training, simple fine-tuning, or continual learning. Re-training considers the complete annotated dataset anew at each iteration and thus allows for a clearer comparison between iterations, characterised by using different quantities of data. However, re-training is time- and resource-consuming, especially with the current

trend of using exceptionally large datasets and neural networks, making it less practical. In contrast, fine-tuning involves modifying the weights of an existing model to fit a new dataset or task, requiring less time and resources compared to re-training. Continual learning techniques can also be employed to gradually improve the model's performance without the need for extensive re-training. However, a method for predictor update is not within the scope of our contributions, and fine-tuning allows for a simple functional predictor update method, which can be an improvement over re-training in optimizing the resource usage, making it more adapted to practical use. Therefore, we choose fine-tuning for the advanced version of SAIM, but ablate the predictor update method and report the experiment results using re-training as well.

5.4 Experimental Results

5.4.1 Instantiation of SAIM and General Setup

We instantiate our system with an interactive predictor, which has two heads. The network is an existing encoder-decoder architecture featuring RNN modules (Sherstinsky, 2020). Specifically, we use a ResNet34 (He et al., 2016) encoder pre-trained on ImageNet (Deng et al., 2009) and a decoder equipped with a pair of standard convolutional layers and a matching convolutional LSTM layer at every step of the upsampling path as presented in section 4. The architecture of the heads is configured as shown in figure 5.4: (1) the LPH branches from the penultimate decoder layer, and (2) the target head is simply the ultimate decoder layer. LPH is trained in sync with the rest of the network. Training configuration and parameters are reported in table 5.3. We do not stop the LPH gradient propagation to the target model as in the original work (Yoo and Kweon, 2019). To counter dataset imbalance, we use the focal loss (Lin et al., 2017) with per-class weights, which are re-calculated prior to re-training at each iteration. We preprocess all data via normalisation, standardisation, and perform random data augmentation: vertical and horizontal flipping, intensity shifting, gamma correction, blurring and unsharp masking. N4BFC (Tustison et al., 2010a) is used for MRI data. The main libraries, the OS and their corresponding versions we use are reported in table 5.5.

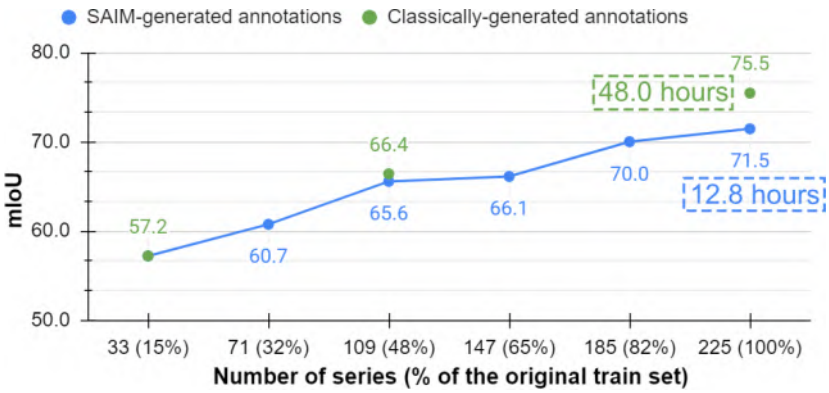


Figure 5.7: `SAIM-base` experimental evaluation results given as mIoU at each iteration on Pancreas CT dataset, where: green - performance using annotation produced using classical tools, blue - performance using SAIM-created annotations. Annotation time is estimated excluding pre-training data annotation with classical tools.

5.4.2 Emulated Annotation Scenarios

General considerations. We perform a systematic evaluation of SAIM's performance in three emulated annotation scenarios: (1) in FPMRI segmentation on our dataset, (2) and (3) in Liver and in Pancreas CT segmentation on decathlon datasets (Antonelli et al., 2022). Two factors make these scenarios emulated: first, these datasets were already fully annotated with high reliability using classical tools; second, human user involvement is not feasible due to the high number of series to annotate. We thus use a virtual user to operate the interactive predictor by simulating user interactions from existing annotations, as during training, done in section 4.

The simulated interactions are clicks, fixed to 3 per class and per image. For each task we proceeded as follows: first, we took a small subset of the annotated data and split it into three parts - the initial training set to pre-train the predictor, the validation and test sets, which remain fixed; second, we alternated between virtual data annotation and predictor update until all data was annotated, while reporting mIoU on the test set at each iteration. We also compared these results to those of a predictor trained with the same quantity of data annotated classically at the first, middle and last SAIM iterations.

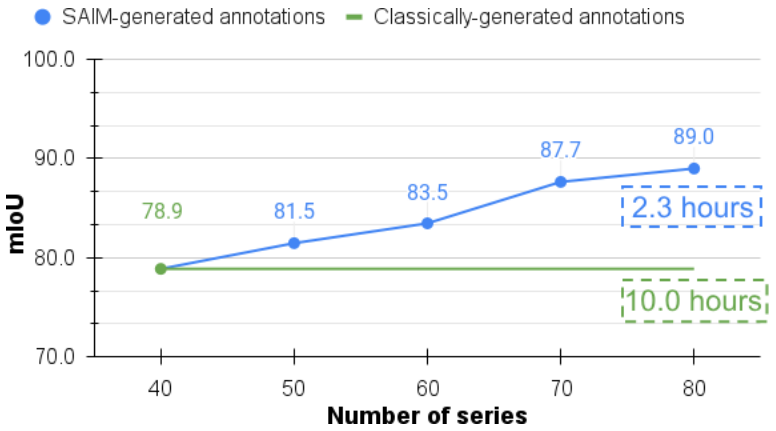


Figure 5.8: `SAIM-base` experimental evaluation results given as mIoU at each iteration on AMOS dataset, where annotations are done by human user via a specifically-developed GUI: green - performance using annotations produced using classical tools, blue - performance using SAIM-created annotations. Annotation time is reported excluding pre-training data annotation with classical tools.

A key factor in these experiments is the initial training set size, which is task- and data-dependent. Since the interactive predictor should produce at least partial annotations, it makes sense to establish the initial training set size as a function of the predictor performance, which is measurable on the test set. For each dataset we pre-trained multiple predictors and selected the one which satisfies two criteria: (1) the IoU score is above 50% for all classes or, if impossible, the **m**IoU is above 50%, and (2) the quantity of data used for pre-training is as low as possible. In a real annotation scenario this additionally depends on the availability and performance of the expert. Therefore, we made the data selection size N as reasonably close to the size of the initial training set as possible, while keeping the overall number of iterations such that a meaningful performance change could be observed in-between. For all datasets we reported mIoU on the test set at each iteration and compared these results to those of a classical system. We also estimated the annotation time using SAIM against 3D Slicer for all datasets.

Female pelvis MRI. We use a FPMRI segmentation dataset FPMRId presented in section 3.1. It

consists of 97 MRI series with 3066 slices in total, manually annotated in 3D Slicer and in MITK by expert radiologists. It involves five classes: `uterus`, `bladder`, uterine `cavity`, `tumours` and `background`. The segmentation of anatomical structures of the female pelvis is particularly challenging due to a large variance in their representation. The dataset is strongly imbalanced due to the anatomical differences between the classes. The original dataset split between the training, validation and test sets is respectively: 77 series (2449 slices), 10 series (308 slices) and 10 series (309 slices). We pre-train the interactive predictor on 15% (12 out of 77 series) of the training set, which achieves 54.6% IoU on a fixed test set. We use this predictor to annotate the remaining 85% (65 series) in 6 iterations, adding 12 series at each of the first 5 iterations and 5 at the last iteration.

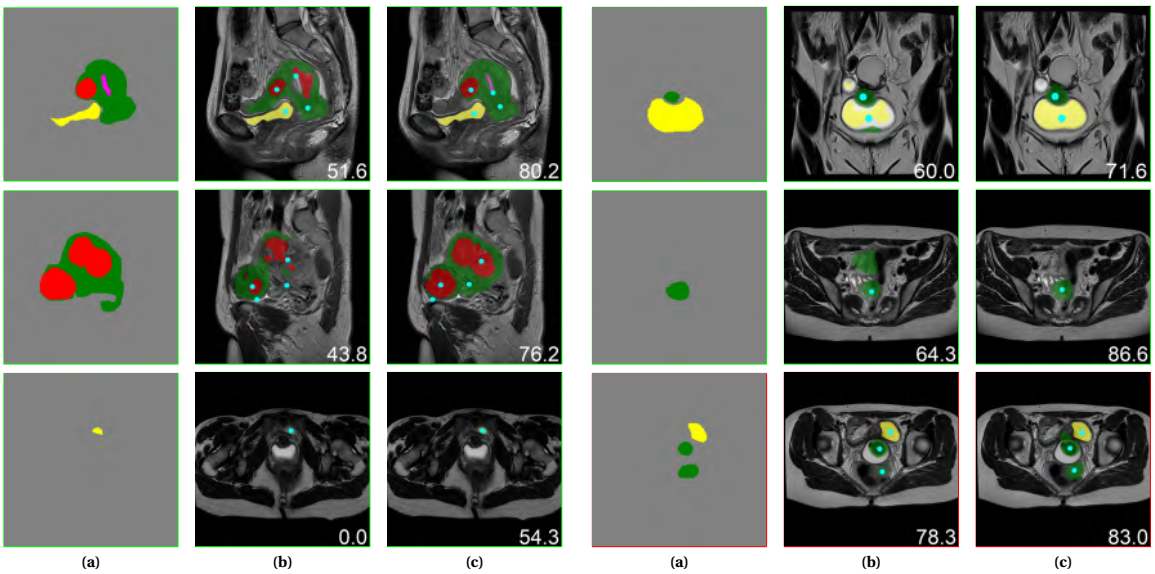


Figure 5.9: `SAIM-base` segmentation results in a emulated annotation scenario for FPMRId, where `uterus` - green, `bladder` - yellow, `tumours` - red, uterine `cavity` - pink, and user clicks are in cyan: **(a)** ground truth; **(b)** interactive predictor pre-trained on 15% (12 out of 77 series) of the complete training set **(c)** the same interactive predictor using the complete training set (77 series), but with the remaining 85% (65 out of 77 series) annotated as a part of SAIM. Performance-wise: rows in green (1-5) - improvement, row in red (6) - considered a degradation despite overall higher IoU due to the false positive for `uterus`. IoU is given in the bottom right corner.

The performance steadily increases with each iteration. The largest change of 8.5pp IoU is observed between iterations 1 and 2 with 57.5% IoU and 66.1% IoU respectively. At 62% of the dataset annotated, SAIM achieves 69.2% IoU against 72.3% IoU for a classical system, which becomes 73.4% IoU against 75.8% IoU at 100% of data. In both cases SAIM slightly underperforms, which is expected since annotation with classical tools is anticipated to have a naturally higher precision. The metrics are reported in figure 5.5. While the performance of a classical annotation system is slightly higher in terms of accuracy, the above results show that with only 47% of data being annotated, SAIM achieves 87% of performance of this classical system trained with all 100% of the data. Crucially however, the annotation time for the whole dataset is decreased by 65%: with SAIM it takes 11.0 hours, including the time spent with 3D Slicer to annotate the initial training set, against 32.0 hours when 3D Slicer is used for the complete dataset. This shows the high impact of using SAIM in this context.

The segmentation results are shown in figure 5.9 and visually demonstrate the performance of the interactive predictor at the first and final SAIM iterations on samples from 6 different series. We use a fixed evaluation set to compare: (b) an interactive predictor trained only on annotations

produced by classical tools (12 series - 15% of the complete training set); (c) the same interactive predictor using the complete training set (77 series) with all remaining annotations produced by the self-same predictor during SAIM iterations. Since in this dataset each class in a single image may be represented by multiple completely disconnected regions, user input is limited to 1 click per region to clearly demonstrate SAIM performance with minimal user input. We observe that in most of the cases additional data annotated using SAIM allowed to substantially improve accuracy, as supported by the metrics in figure 5.5. However, deterioration of accuracy can be observed in a limited number of cases, which we attribute to a shift in the training set between its partial and complete versions.

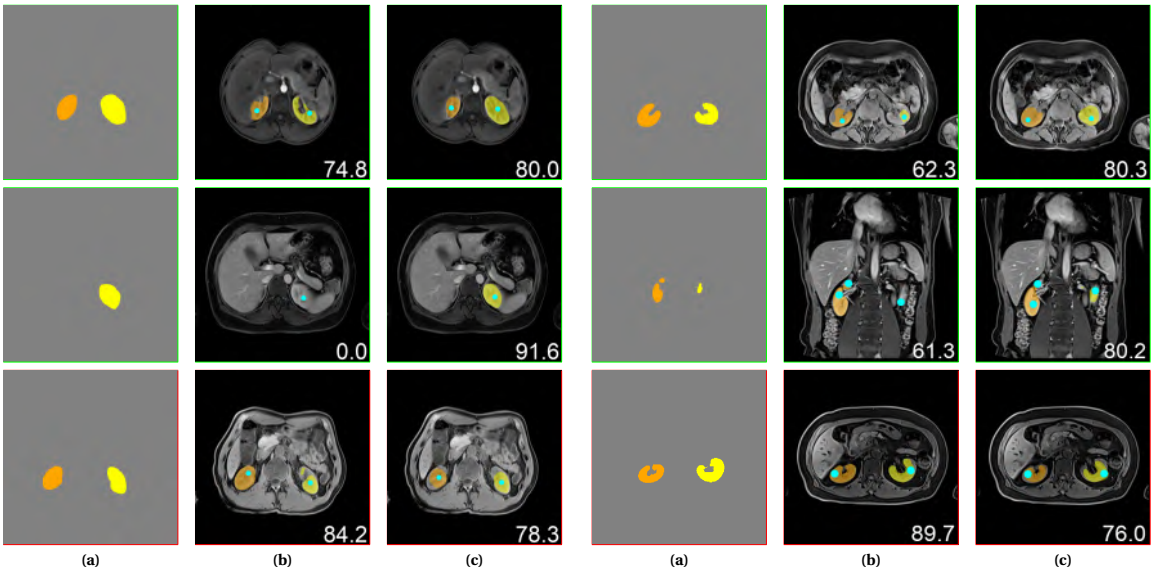


Figure 5.10: `SAIM-base` segmentation results in a real annotation scenario for kidney MRI segmentation on the AMOS dataset, where `right kidney` and `left kidney` are orange and yellow respectively with user clicks in cyan: **(a)** ground truth; **(b)** interactive predictor pre-trained on original AMOS training set (40 series) **(c)** the same interactive predictor after doubling the training set as a part of SAIM (80 series). Performance-wise: rows in green (1-4) - improvement, rows in red (5-6) - degradation. IoU is given in the bottom right corner.

Pancreas and liver CT. We further evaluate SAIM on the tasks of Pancreas and Liver CT segmentation. The test set ground truth is not available. We thus randomly split the training sets for both liver and pancreas, using 70%/15%/15% for training, validation and test respectively, resulting in 91/20/20 series for the liver and 198/42/42 series for the pancreas. Liver CT targets are `liver` and `cancer`, and pancreas CT targets are `pancreas` and `mass` (cyst or tumour). These datasets where annotated manually using classical tools, but the exact software and elapsed time are not specified (Antonelli et al., 2022). We thus record the elapsed time from re-annotating randomly selected series in 3D Slicer, which is extrapolated to obtain the estimates.

Pancreas CT. We pre-train the interactive predictor on 15% of the training dataset (33 out of 225 series), which achieves 57.2% IoU. We use this predictor to annotate the remaining 85% (192 series) in 5 iterations, adding 38 series at each of the first 4 iterations and 40 at the last iteration to have the whole dataset annotated. The performance increases with each iteration with the largest change of 4.9pp IoU between iterations 1 and 2 with 60.7% IoU and 65.6% IoU respectively. At 48%

annotated data, SAIM marginally underperforms compared to a classical system with 65.6% IoU against 66.3% IoU, which is more notable at 100% of data with 71.5% IoU against 75.5% IoU. The metrics are reported in figure 5.7.

While SAIM shows a lesser performance growth between iterations on Pancreas CT compared to the FPMRId, this case still demonstrates that it is possible to annotate the whole Pancreas CT dataset using a predictor initially pre-trained on only 33 series. We attribute the uneven performance growth to the difficulty of distinction between the pancreas and the mass: at 48% annotated data they are at 71.9% and 59.2% for SAIM against 71.1% and 61.7% for a classical system respectively. Still, it is observed that data selection allows SAIM to achieve 87% of the performance of a classical system with 48% of data used, which demonstrated the feasibility of using SAIM in this case.

Table 5.6: Ablation study setups. The tick marks indicate the present architecture elements. One component is rotated out at any time with the exception of `SAIM`, which is the advanced version with all the newly introduced components present, and `SAIM-base` with none of them. The exact features of each version are shown in table 5.1.

Name	**Component**		
	Fine-tuning	LPH	Class-weighting
`SAIM-advanced`	✓	✓	✓
`SAIM-no-fine-tuning`		✓	✓
`SAIM-no-weighting`	✓	✓	
`SAIM-no-loss-criterion`	✓		✓
`SAIM-base`			

Liver CT. Applying SAIM to the Liver CT dataset allows us to start the annotation from pre-training on only 5% of the dataset (5 out of 105 series) with the initial performance at 69.7% IoU. We use this predictor to annotate the remaining 95% (100 series) in 7 iterations, adding 5 series at each of the first 2 iterations and 18 at each of the remaining iterations. The metrics are reported in figure 5.6. The performance grows significantly between that of the pre-trained model and iterations 1 and 2, which is an added 8.8pp and 7.1pp IoU respectively. Notably, it is enough to annotate 14% of data for SAIM to achieve 98% of the classical system's performance with all data. However, performance at iterations 3-7 fluctuates between 84.1% and 85.9%, stopping at the latter and slightly underperforming against 87.0% for a classical system. We attribute these fluctuations to SAIM being already very close to the best achievable performance

5.4.3 Real Annotation Scenario

We demonstrate SAIM in a real annotation scenario for kidney MRI segmentation on the AMOS dataset, which involves three classes: `left kidney`, `right kidney` and `background`. It differs from the emulated scenarios: first, in AMOS only 100 series out of 1200 are annotated, owing to the unfeasible expert effort required; second, a human user operates the interactive predictor for the data annotation step via a developed GUI, for which the interaction number is not limited, but the elapsed time is reported. The AMOS dataset contains both annotated and non-annotated

data, with 100 and 1200 MRI series respectively, collected from multi-centre, multi-vendor, multi-modality, multi-phase, multi-pathology patients. To the best of our knowledge the 1200 MRI series were never annotated previously, owing to the unfeasible expert effort it would require. The annotated data is originally split in 40/20/40 series for the training, validation and test sets respectively. Test annotations are unavailable. We thus leave the training set unchanged and split the validation set, with the new split being 40/10/10 series respectively.

To demonstrate SAIM we pre-trained the interactive predictor on the available 40 series annotated using classical tools and then proceeded using SAIM to double the dataset size. A total of four iterations was performed, each adding 10 series annotated by the human user via the GUI. The performance steadily increases with each iteration, with the largest improvement being 4.2pp IoU between iterations 2 and 3. Overall, with 40 new annotated series the performance on the evaluation set increased by 10.1pp IoU, showing SAIM's efficiency when interactively annotating data from a large and varied non-annotated pool with the help of data selection. SAIM improved the performance of the interactive predictor by 10.1pp IoU from 78.9% IoU to 89.0% IoU. This is further reinforced by the time gain. Specifically, it takes 2.3 hours for SAIM and 10.0 hours estimated for classical annotation tools in 3D Slicer to double the size of the AMOS training set from 40 to 80 series. With SAIM, a single series took 3'43" against 15' on average, significantly decreasing the annotation time, all the while iteratively contributing to SAIM's predictor improvement. The metrics are reported in figure 5.8.

The segmentation results are shown in figure 5.10. We use a fixed evaluation set to compare: (b) an interactive predictor trained only on annotations produced by classical tools (40 series - original AMOS training set); (c) the same interactive predictor after doubling the training set with all new annotations produced by the self-same predictor during SAIM iterations (80 series: 40 series - original AMOS training set and 40 series - newly annotated data). The user interactions as clicks were limited to 3 per image. We observe that in most of the cases additional data annotated using SAIM allowed to improve accuracy, as supported by metrics in figure 5.8. As in section 5.4.2, we attribute the limited number of degradation cases to the training set shift.

5.4.4 Ablation Study

We perform a complete ablation study of the SAIM framework architecture on FPMRId. For this, we begin with the advanced version, denoted `SAIM-advanced` and rotate out each architecture advancement in turn. We also remove all of newly introduced components at once, which results in a base solution, denoted `SAIM-base`, also presented in section 5.3 and published in (Mikhailov et al., 2023). This results in 5 setups in total, described in table 5.6. The ablation study is set up as a set of emulated annotation scenarios, where virtual user operates the interactive predictor with the same rules as in section 5.4.2 for FPMRI. We also report the elapsed mean and total time required for re-training against fine-tuning across all iterations. The complete results are reported in figure 5.11. We first begin with an overview of these results, followed by a detailed iteration-by-iteration analysis, providing our reasoning for the patterns observed. Finally, we discuss the time efficiency of the SAIM predictor update mechanisms, as reported in figure 5.11, and present our conclusions.

Overall, we observe that `SAIM-base` is generally outperformed by all advanced setups until 78% of the dataset (60 series) is annotated. The setups, ordered from the largest to the

smallest performance improvement over SAIM-base, are as follows: SAIM-no-fine-tuning, SAIM-advanced, SAIM-no-weighting and SAIM-no-loss-criterion, where the latter provides marginal improvement. The exception to this rule is SAIM-no-loss-criterion, which shows the lowest performance among all setups at 31% of the dataset annotated with 57.19% IoU, which is slightly below 57.5% of SAIM-base. Starting at 78%, the balance shifts, which leads to SAIM-no-fine-tuning and SAIM-base outperforming the other 3 setups at 100%, with 73.92% and 73.4% mIoU respectively. Furthermore, among these two, SAIM-no-fine-tuning consistently and notably outperforms SAIM-base, as well as all other advanced setups across all iterations, for which varying degrees of improvement are observed.

We now discuss these results iteration by iteration starting with the pre-training done on 15% of the training dataset (12 out of 77 series).

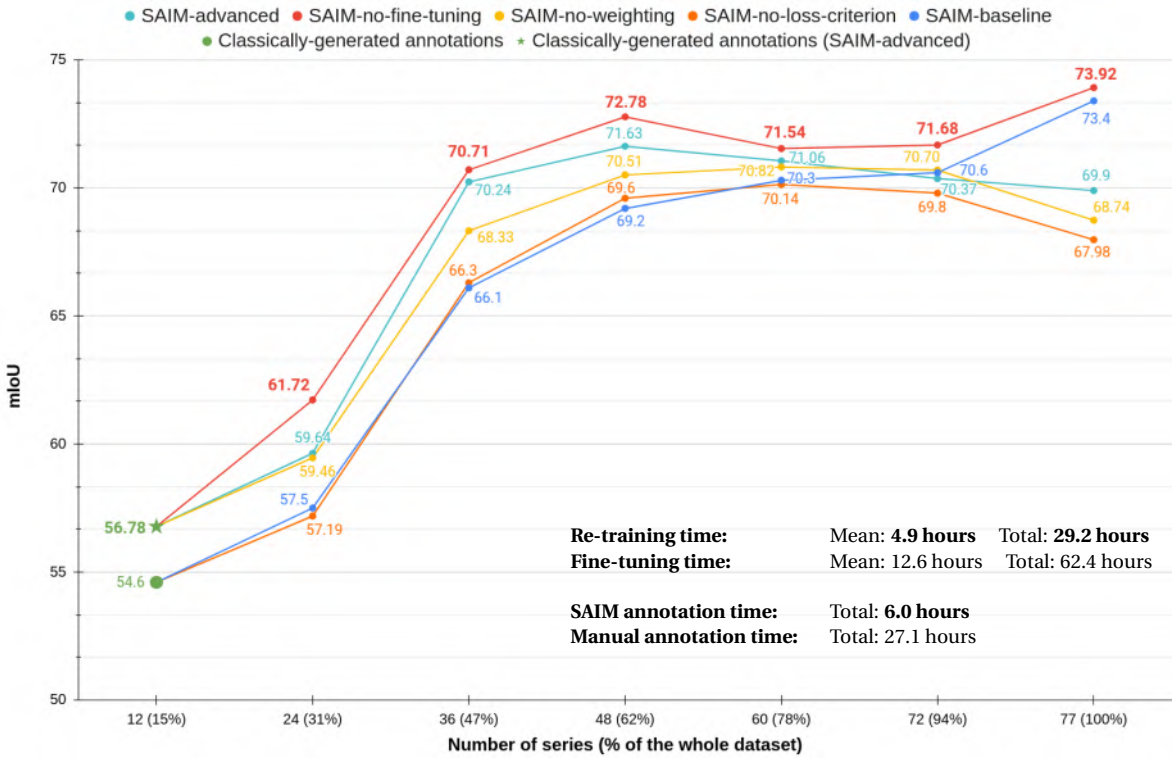


Figure 5.11: SAIM ablation results with 5 setups (see table 5.6) given as mIoU at each iteration, where green - performance using annotations produced using classical tools. First iteration features two results for SAIM-base and SAIM-advanced predictors, which are represented by a circle and a star respectively. Difference in performance for the latter is due to the addition of LPH in SAIM-advanced. For other iterations, we compare these ablation results with those of a predictor trained on classically-generated annotations only in the zoomed-in version of this graph in figure 5.12. In the bottom right, we report the predictor update times for both re-training and fine-tuning, presented as the mean and total across all iterations. Additionally, the total annotation times for manual annotation and SAIM are provided. Best results are highlighted in bold.

15% (12 series). For this experiment, the pre-training is done twice: once for SAIM-base and SAIM-advanced, which achieves 54.6% and 56.78% mIoU respectively on a fixed test set. SAIM-no-loss-criterion has matching performance with SAIM-base, since introduced additional class-weighting does not affect pre-training. We attribute the increase in performance for SAIM-advanced to the introduction of the (LPH) - a second head, as it enables the model to benefit from an auxiliary task of loss prediction that seems to enhance representation learning.

31% (24 series). By 31% of the dataset, all setups incorporating the loss criterion show a marked performance increase with `SAIM-no-fine-tuning` achieving best performance. We assume that this performance improvement is all the more pronounced due to the inherently higher starting point of the non-`SAIM-base` predictor. However, `SAIM-no-loss-criterion` falls below `SAIM-base`. This might suggest that at this iteration fine-tuning is detrimental to predictor's performance as evidenced by the initially higher starting performance of this predictor, notably higher performance of `SAIM-no-fine-tuning`, as well as a minimal difference between `SAIM-advanced` and `SAIM-no-weighting`. We attribute this to the predictor's inability to effectively annotate and capture the features of the weighted selected data. Simply, weighted data selection is more challenging, since it targets bringing in smaller classes (e.g. `cavity`), which the predictor, at its current stage, may struggle to handle with high accuracy.

47% (36 series). At 47% of the dataset, class-weighting emerges as a major performance improvement factor with `SAIM-advanced` improving over `SAIM-no-weighting` with 70.24% against 68.33% mIoU respectively. While `SAIM-no-loss-criterion` shows very minor improvement over `SAIM-base`, loss criterion presence remains the biggest driving force behind performance improvement. This is evidenced by `SAIM-no-fine-tuning` remaining the best-performing setup with 70.71% mIoU. We attribute this effectiveness of class-weighting as compared with previous iteration to the growing capability of the predictor to effectively segment the whole range of classes present in FPMRId.

62% (48 series). At 62% of the dataset, all setups show improvement compared to the previous iteration. However, at the same time the results arrive at a plateau, where further performance gains become challenging. For example, `SAIM-advanced` peaks at 71.63% and only loses performance during the next iterations. `SAIM-no-fine-tuning` remains the best-performing setup with `SAIM-advanced` remaining second best. We compare these results to that of two predictors trained only on classically-annotated data in figure 5.12, which zooms onto the range of results between 47% and 100% of annotated data. The goal is to observe the performance influence of SAIM-produced annotations on the predictor as compared to manual annotations. Specifically, we train two predictors using the training sets of `SAIM-base` and `SAIM-no-fine-tuning`, which are different due to two distinct data selection policies, but with all data annotated classically. We note that `SAIM-no-fine-tuning` achieves 72.78% mIoU, while `SAIM-base` and `SAIM-no-fine-tuning` predictors with classical annotations achieve 72.3% and 73.3% mIoU respectively. Effectively, with `SAIM-no-fine-tuning` the difference between training on classically- and SAIM-annotated data is further reduced from 3.1pp (for `SAIM-base` previously) to 1pp IoU, showing the advantage of loss criterion and class weighting employed in `SAIM-no-fine-tuning`.

78% (60 series). As we move to 78%, we observe slight decline in performance for 2 setups and similarly slight improvement for 3. Among the latter are `SAIM-no-fine-tuning` and `SAIM-advanced`, while the former are `SAIM-no-weighting`, `SAIM-base`, and `SAIM-no-loss-criterion`. As stated previously, we observe a plateau in performance for iterations from 62% to 94% of the dataset, which we attribute to two main factors. First, the actual performance of all setups is close to the maximum observed performance achieved with classically-annotated

data shown in figure 5.12. Second, the two data selection criteria we employ are not inherently designed to guarantee performance improvements on the fixed testing set, in accordance with scientific integrity principles to avoid bias. In simpler terms, while the selected data is the most informative for the predictor, the predictor may not always choose data that optimizes performance on this specific testing set at each iteration. This ensures unbiased results, but may prevent achieving the best possible improvement on the testing set. As a result, performance plateaus are a real possibility. Addressing this requires a larger and more diverse testing set, which is a persistent challenge in medical imaging due to limited accessibility. At the same time, the curation and creation of a representative testing set in medical imaging is a research area of its own and falls outside the scope of this work.

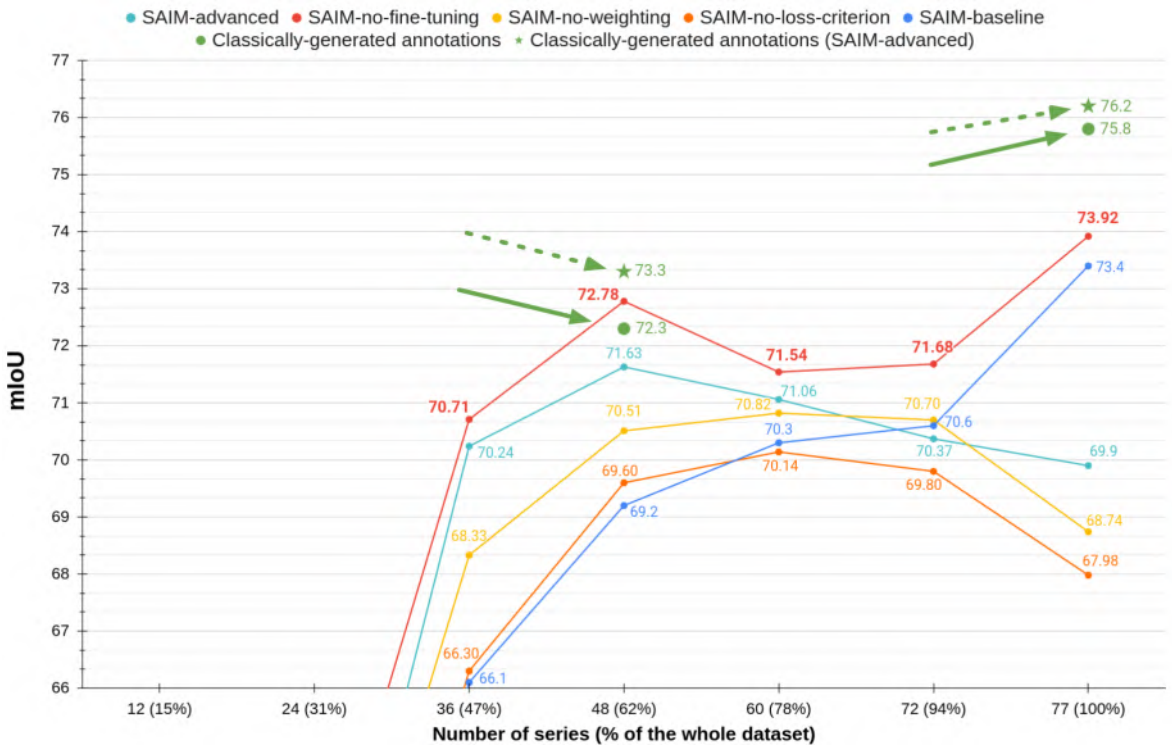


Figure 5.12: Comparison of the ablation results presented in figure 5.11 (zoomed-in) with those of predictors trained on classically-generated annotations, where green - performance using annotations produced using classical tools at 62% and 100% of the FPMRId. Circle and star symbols represent the `SAIM-base` and `SAIM-advanced` predictors, indicated by solid and dashed arrows, respectively. Difference in performance is both due to the addition of LPH and a different data selection criterion in `SAIM-advanced`, which results in a training set mismatch between `SAIM-base` and `SAIM-advanced`. Best results are highlighted in bold.

94% (72 series). The performance plateau continues at 94% of the dataset. However, compared to previous iteration, we observe slight improvement for `SAIM-no-fine-tuning`, while `SAIM-no-weighting` and `SAIM-no-loss-criterion` begin to decline. Overall, only `SAIM-no-fine-tuning` and `SAIM-base` demonstrate improvement in this iteration, scaling up this positive trend into the next.

100% (77 series). At 100% of the dataset we observe that `SAIM-no-fine-tuning` and `SAIM-base` demonstrate overall best performance with 73.92% and 73.4% mIoU respectively. As shown in figure 5.12, these predictors achieve results within 2.28pp and 2.4pp mIoU respectively from their

counterparts trained on classical annotations, showing a slight discrepancy reduction for `SAIM-no-fine-tuning`. We attribute the notable performance improvements of both `SAIM-no-fine-tuning` and `SAIM-base` over the previous iteration to data selection methods indirect influence on the fixed testing set. Simply, as discussed in relation to the setups' performance at 76% of the dataset, certain series with the potential to improve performance are overlooked by the data selection policies. These series are consistently not chosen for annotation by the predictors, which may limit the overall performance gains. This results in a performance improvement spike when these series are finally annotated and added to the training set.

We observe that setups utilising fine-tuning, as opposed to those relying on re-training, exhibit a more pronounced decline in performance during this iteration. We explain this gradual decrease in performance, which began as early as the iteration at 62% of data for `SAIM-advanced`, by catastrophic forgetting. This phenomenon causes the performance of the neural network to degrade due to its inability to retain previously learned information during continuous predictor updates. In contrast, re-training the model by treating each iteration's data as a complete dataset to train on from scratch, avoids this issue and maintains performance consistency. This results in `SAIM-no-fine-tuning`, which features re-training as the method of predictor update, to achieve overall best performance in this ablation study.

Time efficiency: Predictor Update. Although the literature acknowledges that fine-tuning generally requires less time than retraining, particularly with large datasets, we demonstrate that this advantage may not hold in the continual learning setting when dealing with moderate amounts of data. As shown in figure 5.11, `SAIM-no-fine-tuning` achieves more than twice the speed in both mean and total convergence during retraining compared to fine-tuning, with times of 4.9 and 29.2 hours against 12.6 and 62.4 hours, respectively. We attribute this to the challenging nature of medical image segmentation, exacerbated by low data regime and lower predictor performance during the first iterations, which result in an increased convergence time when doing fine-tuning. Simply, starting fresh might converge faster than adapting an existing trained model to a larger data distribution by introducing edge cases via data selection.

Conclusion. The ablation study highlights four key insights into the performance of the SAIM framework. First, the inclusion of the loss criterion with LPH emerges as the most significant factor driving performance improvements, consistently enabling superior results across iterations. Second, class-weighting also plays an important role, particularly starting intermediate iterations. However, it may provide close to no improvement or potentially harm the performance, when the predictor is not yet capable of segmenting complex under-represented classes, such as `cavity` in FPMRId. Third, while fine-tuning offers marginal iteration-to-iteration gains in a limited number of iterations, it suffers from catastrophic forgetting over time, making re-training a more effective strategy for maintaining performance consistency. Lastly, the entropy and loss-prediction-based data selection strategies occasionally fail to select critical, representative images from the dataset, resulting in a performance plateau between 62% and 94% of the dataset annotated. Still, this plateau is, on average, only 5.43pp mIoU below the best result achieved with 100% of the data manually annotated. However, once these overlooked images are annotated and incorporated in later stages, they lead to marked performance improvements, reducing the performance gap between SAIM-annotated and classically-annotated data to a minimum of 2.28 pp mIoU for `SAIM-`

`no-fine-tuning`, as demonstrated in the final iteration. Combining multiple data selection policies, incorporating the internal and external criteria, both supported by SAIM, could potentially serve as a starting point to mitigate this problem.

5.4.5 SOTA Comparison

We compare SAIM with methods from the two most relevant domains - SSL and ST, represented by 5 SSL approaches on ACDC dataset with 70 series in total, as well as by 4 ST and 3 SSL approaches on SBD-augmented Pascal VOC with 10582 images in total. This is the first evaluation of SAIM on these datasets. The comparisons are setup as emulated annotation scenarios, where the virtual user operates the interactive predictor with the same rules as in sections 5.4.2 and 5.4.4. For this experiment, we maintain the architecture detailed in Section 5.4.1, but replace the backbone with ResNet50 (He et al., 2016) instead of ResNet34 to be on par with other methods.

Method	**Classically-generated annotations**		
	1/16 (662)	1/8 (1323)	1/4 (2645)
Supervised-baseline (Yang et al., 2022)	64.0	69.0	71.7
CCT (Ouali et al., 2020)	65.2	70.9	73.4
CutMix-Seg (French et al., 2019)	68.9	71.7	72.5
GCT (Ke et al., 2020)	64.1	70.5	73.5
CPS (Chen et al., 2021c)	68.2	73.2	74.2
CPS† (Chen et al., 2021c)	72.0	74.3	74.9
ST (Yang et al., 2022)	72.2	74.8	75.5
ST++ (Yang et al., 2022)	73.2	75.5	76.0
SAIM (ours)	**78.3**	**79.2**	**80.4**

Table 5.9: Experiment results for SAIM in realistic scene image segmentation against 4 ST and 3 SSL methods on Pascal VOC dataset given as mIoU. Listed methods' domain attribution is as follows per-line : (1) supervised baseline, (2-4) SSL and (5-8) ST. The final scores are provided for models trained with all available data. The 'Classically-generated annotations' column indicates the fraction of data paired with the original Pascal VOC annotations. For the remaining data, the methods differ: SAIM and ST approaches use annotations generated by themselves, while SSL methods utilise data without annotations.

ST. We compare SAIM with (Yang et al., 2022) in realistic scene image segmentation on Pascal VOC dataset, which involves 21 class, including `background`. The results are presented in table 5.9 and vary based on the subset of the original classically annotated dataset used as input. Specifically, the subsets comprise 1/16, 1/8, and 1/4 of the entire dataset. Dataset splits are the same for all methods. We observe that SAIM achieves the best performance no matter the subset used for pre-training with ST methods being the closest. It can be mainly attributed to SAIM capability to produce strong instead of the pseudo- labels used in ST methods. Although ST++ attempts to address label reliability by identifying and prioritising reliable labels for training, it still falls short.

SSL. We compare SAIM with (Yang et al., 2023a) in cardiac MRI segmentation on ACDC dataset, which involves 4 classes: `Right Ventricular (RV) cavity`, `myocardium`, `Left Ventricular (LV) cavity` and `background`. The results are presented in table 5.11 and vary based on the size of the classically-annotated subset used as input. Specifically, the subsets consist of 1, 3, and 7

series. Dataset splits are the same for all methods. We observe that SAIM achieves the best overall performance, but with small margins. For instance, using the 1-, 3-, and 7-series subsets, SAIM achieves Dice scores of 86.0, 90.3, and 91.4, outperforming UniMatch by 0.6pp, 1.4pp, and 1.5pp respectively. We attribute this to the two main factors: (1) the limited pre-training dataset sizes with only 32 images in the 1-series subset, and (2) the performance already nearing the upper bound achieved by recent methods using only classically annotated data, such as (Kato and Hotta, 2024).

Method	**Classically-generated annotations**		
	1 series	3 series	7 series
Supervised-baseline (Yang et al., 2023a)	28.5	41.5	62.5
UA-MT (Yu et al., 2019)	N/A	61.0	81.5
CPS (Chen et al., 2021c)	N/A	60.3	83.3
CNN & Trans (Luo et al., 2022)	N/A	65.6	86.4
UniMatch (Yang et al., 2023a)	85.4	88.9	89.9
SAIM (ours)	**86.0**	**90.3**	**91.4**

Table 5.11: Experiment results for SAIM in cardiac MRI segmentation against 4 SSL methods on ACDC dataset given as Dice. The final scores are provided for models trained with all available data. The 'Classically-generated annotations' column indicates the fraction of data paired with the original ACDC annotations.

5.5 Conclusion

We have proposed a general concurrent neural predictor training and data annotation framework called SAIM. The strength of SAIM is its unique ability to exploit the newly annotated data as the annotation task progresses in order to improve the annotation mechanism. This is achieved by involving the predictor being trained in the steps of data selection and of interactive annotation. The neural model is thus always up-to-date and coherently shared by all the system components, contributing to optimal choices and quick improvements of the predictor performance as annotation proceeds. As a consequence, SAIM allows one to annotate massive datasets fast from very limited initial annotations.

We evaluated SAIM in five emulated annotation scenarios using fully-annotated segmentation datasets, including FPMRI segmentation, using FPMRId dataset, liver and pancreas CT segmentation from the medical segmentation decathlon challenge, cardiac MRI segmentation on ACDC dataset, and natural image segmentation on Pascal VOC 2012 dataset. Furthermore, we compared SAIM to state-of-the-art approaches in the closest domains to our contribution, evaluating it against 10 ST and SSL methods in total. We also conducted an ablation study to evaluate the impact of individual components in SAIM by comparing base and advanced versions of the framework. The advanced version incorporates two key improvements over the base version: (1) LPH, which introduces state-of-the-art data selection criterion based on predicted loss and (2) class-based weighting, which address class-imbalance issues when selecting data. The study demonstrated that these enhancements significantly boost SAIM's performance while highlighting the time efficiency of re-training over fine-tuning for smaller datasets, with re-training achieving more than twice the convergence speed. We also applied SAIM to AMOS kidney MRI segmentation - a

real-world case of very large dataset annotation, which cannot be feasibly annotated otherwise. This shows that SAIM jumpstarts efficient interactive annotation from limited annotated data and minimises the amount of data to annotate, while improving predictor performance. Simply, SAIM is a powerful two-in-one annotation and training solution to drag-and-drop in a large dataset annotation task without the need for an efficient neural predictor to be prepared first.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Image segmentation remains challenging despite active research and numerous proposed solutions. This is especially true in the medical domain, where limited data, such as FPMRI, variations in image quality, organ morphology, and the presence of diverse pathologies make it challenging for both radiologists and computational approaches. At the same time, existing methods often overlook the crucial role of expert involvement in clinical setting, and the substantial resources necessary for medical image annotation compared to natural images. Still, clinically-adapted methods that effectively address these challenges are crucial to reduce the radiologists' workload and streamline both the prototyping and deployment of new AI solutions. In this work, we introduced four distinct contributions to segmentation with two data-wise and two application-wise contributions respectively. On the data side, we established a new FPMRI segmentation dataset called FPMRId and investigated its inter-expert variability to lay the groundwork for robust segmentation model development. On the application side, we presented two key contributions designed with industrial and clinical usage in mind. First, a framework for interactive segmentation, which takes in consideration the order of user interactions to improve performance. Second, a general annotation framework effective in limited annotated data regimes, featuring a single model shared between the tasks of annotation, training and data selection.

On the data side, we are gathering and curating what is, to the best of our knowledge, the first large-scale FPMRI dataset currently comprising 374 medical scans with segmentations for nine classes: (1) bladder, (2) uterus, (3) uterine cavity, (4) cervix, (5) fundus, 6) anterior wall, (7) uterine myomas, (8) endometriosis, and (9) adenomyosis. Five of these classes (1, 2, 3, 4, and 7, respectively) were the focus of our inter-expert variability study, the first of its kind for these classes in FPMRI. The study revealed high agreement for larger, well-defined structures (e.g., uterus and bladder) but moderate and less consistent agreement for smaller, more complex ones (e.g., cervix and uterine cavity). Overall, we observed that while manual segmentations are generally consistent among experts, certain scans may exhibit characteristics which demand special attention, requiring each scan to be examined on an individual basis. These findings informed our subsequent application-oriented contributions by highlighting the importance of expert input during annotation, the challenges posed by limited data, and the need for particular focus on complex and under-represented classes.

Building on this foundation, we proposed an interactive segmentation framework that em-

ploys sequential memory to treat user corrections as a sequence rather than an unordered set of clicks, thereby improving segmentation accuracy with fewer interactions. Our experiments on FPMRI, as well as on liver and pancreas CT scans, demonstrated both performance gains and reduced annotation time for medical experts, making the framework attractive for clinical usage. This framework is intended to become a part of the industrial segmentation solution SURGAR-PLAN by SURGAR (SURGAR, 2024), which is used to construct 3D models from MRI scans for surgical AR.

Further, we introduced SAIM, a framework that integrates data selection, annotation, and model training into a single end-to-end system. This yields a powerful 'two-in-one' solution, which is both an annotation tool and a predictor that can be readily adopted in clinical workflows or industrial pipelines where large-scale data annotation is typically a bottleneck. By incorporating active learning, SAIM prioritizes the most informative samples, thereby minimizing annotation effort while steadily improving the model. We designed SAIM to bridge the gap between data annotation and model deployment, which are often separated in conventional industrial workflows, necessitating additional resources. We find that this principle, alongside model sharing, forms the foundational basis of SAIM. With or without adhering to the architectural specifics detailed in chapter 5, this foundational basis offers considerable benefits in industrial settings and warrants further exploration. Currently, the advanced version of SAIM and its underlying principles are intended for internal use and to be built upon in new annotation projects as outlined in section 6.2.

The results presented in this work indicate that the proposed methods are ready for industrial transfer and are broadly applicable to other medical imaging domains beyond FPMRI segmentation, acknowledging that the industrial transfer is an undertaking in itself. More precisely, the proposed expert-controlled frameworks targeting annotation workload reduction are critical to enable deployment and growth of the diagnostic and decision-support systems featuring DL, such as U-SURGAR (SURGAR, 2024).

6.2 Future Work

In this section, we discuss avenues for improving both the interactive segmentation framework (chapter 4) and SAIM (chapter 5). We organize these potential enhancements into short-term and long-term, further grouped under four main categories: methodology, data, evaluation, and technical, according to the aspect to be improved. This results in six short-term and eight long-term improvements in total.

6.2.1 Short-term

Methodological Limitations

Interactive Segmentation Framework

1. Volumetric segmentation
   The transition from slice-by-slice to volumetric segmentation could allow leverage the 3D nature of medical imaging, thereby potentially improving the capture of anatomical continuity, reducing user interactions, and increasing accuracy. This shift requires adaptation of

the SIM and DDG modules of the framework to volumetric data, as well as employment of a suitable volumetric network architecture.

Data Limitations

Interactive Segmentation Framework

2. Same domain pre-training
   While pre-training on ImageNet is effective, it is widely recognized that performance is further improved when pre-training data is closer to the target domain. To the best of our knowledge, the recently released UMD (Pan et al., 2024) is currently the largest publicly available dataset for uterine MRI segmentation, comprising 300 annotated T2-weighted sagittal images with uterine myomas, as discussed in section 2.2.4. Although UMD is limited in the number of featured classes, pre-training on this dataset may still boost our framework's performance, which is all the more challenging due to severe lack of available annotated data in FPMRI segmentation domain.

3. Training on the up-to-date FPMRId
   FPMRId currently comprises 374 medical scans, of which only 97 are used in this work for training. This subset was selected to maintain consistent experimental evaluation throughout the ongoing annotation process, which runs in parallel with the research. Training on the complete FPMRId could substantially enhance generalization and boost performance.

Evaluation Limitations

SAIM

4. Predictor architecture comparison
   Further experimental evaluation is needed to assess how different network backbones affect SAIM's performance. For instance, integrating Segment Anything Model (SAM) (Ravi et al., 2024), a SOTA segmentation model, into SAIM as the predictor would clarify how the predictor's performance impacts SAIM as a whole.

5. Additional real annotation scenarios
   Extending the current experiments to additional real annotation scenarios would further validate SAIM in the industrial annotation setting for which it is designed. According to chapter 5, a scenario is considered 'real' if two conditions are fulfilled: (1) the dataset comprises predominantly unannotated data with only a small annotated subset, and (2) SAIM is operated by a human expert. Evaluating SAIM under these conditions, rather than relying solely on simulated annotation scenarios that use fully annotated datasets, would provide additional insights into its versatility and performance.

Technical Limitations

Interactive Segmentation Framework & *SAIM*

6. Refactoring and Optimisation
   Both frameworks, currently research prototypes, require refactoring and optimization. The interactive segmentation framework integrates complex input-output management due to

its SIM and DDG components. In turn, SAIM dynamically repartitions annotated and non-annotated data pools, as well as iteratively updating the predictor. Streamlining these processes would speed up annotation, data selection and predictor update processes, as well as prepare a solid foundation for future improvements and maintenance.

6.2.2 Long-term

Methodological Limitations

SAIM

1. Pre-training data size estimation
   An important practical question when using SAIM is how many pre-training data points per task achieve the best balance between initial model performance, performance growth over time, and the resources required. For example, starting with a very large annotated dataset (e.g., millions of images) may yield high early accuracy and permit rapid annotation of new data, but such dataset would be prohibitively time-consuming and costly to produce. Conversely, an extremely small initial dataset reduces the initial classical annotation burden, but limits the starting performance. Producing a module capable of determining the optimal quantity of annotated pre-training data, balancing strong initial predictions against unnecessary resource use, would allow for better estimations and greater control in SAIM-enabled annotation projects.

2. Controlled domain shift
   SAIM inherently induces domain shift by incorporating newly annotated data deliberately chosen for its dissimilarity from what its predictor was trained on. This results in two key avenues for improvement. First, this shift could be monitored and quantified to ensure stable performance as the predictor's learned feature distribution evolves. Second, SAIM can be advanced into a specialized domain adaptation solution guided by data selection through AL. To enable these, it is of interest to potentially extend SAIM capabilities to those of the SOTA approaches. For example, if the predictor was trained with data from one hospital and newly acquired data comes from another, SAIM could enable a controlled domain adaptation process while preserving its performance on the original domain.

Applicative Limitations

Interactive Segmentation Framework

3. Improved background class interactivity
   Based on the user experiment feedback, reducing false positives in the background class is challenging. This is because the network often learns to recognize the background too easily, leading it to place less importance on background clicks during training. Implementing techniques such as BRS, discussed in section 4.5.2, could promote more effective click placement for the background class. Additionally, exploring class-specific regularization based on class complexity may help balance the learning process by penalizing simpler classes more heavily. Although methods like focal loss and per-class weighting offer some improvements, the overwhelming volume of background data limits their effectiveness. Therefore, proposing more advanced regularization techniques is essential to enhance background interaction without negatively impacting the performance on other classes.

Further Improvements

Interactive Segmentation Framework

4. Advanced DDG
   Currently, the click generation mechanism is probabilistic with a fixed maximum number of clicks. Although DDG is based on classes and their components present in each image, it does not fully take into consideration the image's complexity. This could be improved by dynamically adapting the number of generated clicks based on the model's real-time performance when training. A simple solution would be to potentially integrate such performance metric into the loss function. Additionally, exploring various click encodings, such as disc-shaped clicks with or without intensity gradients, as well as other shapes, may result in better performance.

5. Advanced SIM
   SIM is formed by maintaining a sequence of masks containing the user interactions. Future work could investigate alternative memory architectures, such as Space-Time Memory Network (STM), Extended Long Short-Term Memory (xLSTM) and attention-based mechanisms. Specifically, SIM could be adapted to store learned features derived from these masks with or without attention mechanisms. Such an approach offers two key advantages: (1) improved performance - by better capturing the context and dependencies between user interactions, and (2) reduced resource consumption - by storing compressed features instead of the original interaction masks. This reduction in GPU memory usage can potentially allow for larger batch sizes.

6. Extend to other applications
   As demonstrated in the experimental evaluation (section 4.5), our framework is applicable to a wide range of medical image segmentation tasks. Future extensions could target challenging classes within the FPMRId dataset that lack clearly defined contours, including (1) cervix, (2) anterior wall, (3) fundus, (4) endometriosis, and (5) adenomyosis. Specifically, as discussed in section 3.1.2 and illustrated by our inter-expert variability study in section 3.2, classes 1 to 3 have contours that are not consistently identifiable by experts with high precision. Additionally, classes 4 and 5 are known for exhibiting significant variability in their appearance, presenting challenges even for experienced practitioners. Extending our framework to effectively segment these complex and variable classes could enhance its generalisation capabilities and robustness with on existing classes.

7. Reinforcement learning for interactive segmentation
   Reinforcement learning is a promising research area that appears to align with the objective of enabling the network to effectively respond to user interactions, which is being explored in the literature. Simply, by treating each user click as an action that influences the segmentation state, reinforcement learning can more accurately mimic human interaction patterns during training compared to the probabilistic methods used in DDG. This transition has the potential to enhance training efficiency and increase the impact of user interactions at test time.

SAIM

8. Advanced predictor update

   SAIM currently relies on re-training or fine-tuning for predictor update. However, these approaches present certain limitations. Re-training becomes impractical for large and continuously growing datasets due to the increase in computational resources required for each subsequent iteration. In turn, repeated fine-tuning leads to catastrophic forgetting, where the predictor's performance on previously learned data deteriorates as it adapts to new information, as shown in section 5.4. To overcome these challenges, considering SOTA continual learning techniques is essential. Specifically, continual learning could offer two main advantages: (1) reducing the predictor update time, and (2) maintaining stable performance across all data, regardless of its recency. Furthermore, continual learning might allow to adopt a fine-grained update mechanism, such as per series, per slice, per label, or per click. This means immediately utilising validated annotations for real-time model update to accelerate the annotation process for similar subsequent data. At the same time, with this fine-grained approach, new avenues for improvement could open. For instance, it appears beneficial to temporarily overfit a copy of the predictor on specific series, either in an unsupervised or semi-supervised manner before annotation or in a supervised manner during annotation. This targeted approach can improve performance for each specific series being annotated, followed by updating the source predictor with the resulting annotations prior to advancing to the next series.

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. 45

Abadia, A. F., Yacoub, B., Stringer, N., Snoddy, M., Kocher, M., Schoepf, U. J., Aquino, G. J., Kabakus, I., Dargis, D., Hoelzer, P., et al. (2022). Diagnostic accuracy and performance of artificial intelligence in detecting lung nodules in patients with complex lung disease: a noninferiority study. *Journal of thoracic imaging*, 37(3):154–161. 7

Adadi, A. (2021). A survey on data-efficient algorithms in big data era. *Journal of Big Data*, 8(1):24. 20, 29, 111

Adlung, L., Cohen, Y., Mor, U., and Elinav, E. (2021). Machine learning in clinical decision making. *Med*, 2(6):642–665. 50, 51

Aflalo, A., Bagon, S., Kashti, T., and Eldar, Y. C. (2022). Deepcut: Unsupervised segmentation using graph neural networks clustering. *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 32–41. 14

Ahishakiye, E., Bastiaan Van Gijzen, M., Tumwiine, J., Wario, R., and Obungoloch, J. (2021). A survey on deep learning in medical image reconstruction. *Intelligent Medicine*, 1(03):118–127. 19

Al-Qatf, M., Lasheng, Y., Al-Habib, M., and Al-Sabahi, K. (2018). Deep learning approach combining sparse autoencoder with svm for network intrusion detection. *Ieee Access*, 6:52843–52856. 7

Albahri, A. S., Duhaim, A. M., Fadhel, M. A., Alnoor, A., Baqer, N. S., Alzubaidi, L., Albahri, O. S., Alamoodi, A. H., Bai, J., Salhi, A., et al. (2023). A systematic review of trustworthy and explainable artificial intelligence in healthcare: Assessment of quality, bias risk, and data fusion. *Information Fusion*, 96:156–191. 18

Albu, A. B., Beugeling, T., and Laurendeau, D. (2008). A morphology-based approach for interslice interpolation of anatomical slices from volumetric images. *IEEE Transactions on Biomedical Engineering*, 55(8):2022–2038. 113

Aljabar, P., Heckemann, R. A., Hammers, A., Hajnal, J. V., and Rueckert, D. (2009). Multi-atlas based segmentation of brain images: atlas selection and its effect on accuracy. *Neuroimage*, 46(3):726–738. 22

Aljabri, M., AlAmir, M., AlGhamdi, M., Abdel-Mottaleb, M., and Collado-Mesa, F. (2022). Towards a better understanding of annotation tools for medical imaging: a survey. *Multimedia Tools and Applications*, 81(18). 116

Alowais, S. A., Alghamdi, S. S., Alsuhebany, N., Alqahtani, T., Alshaya, A. I., Almohareb, S. N., Aldairem, A., Alrashed, M., Bin Saleh, K., Badreldin, H. A., et al. (2023). Revolutionizing healthcare: the role of artificial intelligence in clinical practice. *BMC medical education*, 23(1):689. 48

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., and Farhan, L. (2021). Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8. 3, 111, 112

Amini, M.-R., Feofanov, V., Pauletto, L., Devijver, E., and Maximov, Y. (2022). Self-training: A survey. *ArXiv*, abs/2202.12040. 111, 112, 116

Amrehn, M., Gaube, S., Unberath, M., Schebesch, F., Horz, T., Strumia, M., Steidl, S., Kowarschik, M., and Maier, A. K. (2017). Ui-net: Interactive artificial neural networks for iterative image segmentation based on a user model. In *Eurographics Workshop on Visual Computing for Biomedicine*. 93, 98, 117

Amyar, A., Modzelewski, R., Li, H., and Ruan, S. (2020). Multi-task deep learning based ct imaging analysis for covid-19 pneumonia: Classification and segmentation. *Computers in biology and medicine*, 126:104037. 6

Angelov, V., Petkov, E., Shipkovenski, G., and Kalushkov, T. (2020). Modern virtual reality headsets. In *2020 International congress on human-computer interaction, optimization and robotic applications (HORA)*, pages 1–5. IEEE. 26

Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B. A., Litjens, G., Menze, B., Ronneberger, O., Summers, R. M., et al. (2022). The medical segmentation decathlon. *Nature communications*, 13(1):4128. 14, 89, 101, 114, 125, 127

Aqib, M., Mehmood, R., Albeshri, A., and Alzahrani, A. (2018). Disaster management in smart cities by forecasting traffic plan using deep learning and gpus. In *Smart Societies, Infrastructure, Technologies and Applications: First International Conference, SCITA 2017, Jeddah, Saudi Arabia, November 27–29, 2017, Proceedings 1*, pages 139–154. Springer. 7

Arakelyan, G., Soghomonyan, G., and The Aim team (2024). Aim. 46

Arena, F., Collotta, M., Pau, G., and Termine, F. (2022). An overview of augmented reality. *Computers*, 11(2):28. 25

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*. 13

Bahl, M. (2022). Artificial intelligence in clinical practice: implementation considerations and barriers. *Journal of Breast Imaging*, 4(6):632–639. 48

Baker, M., Jensen, J. A., and Behrens, C. F. (2013). Inter-operator variability in defining uterine position using three-dimensional ultrasound imaging. In *2013 IEEE International Ultrasonics Symposium (IUS)*, pages 848–851. IEEE. 67

Bandi, A., Adapa, P. V. S. R., and Kuchi, Y. E. V. P. K. (2023). The power of generative ai: A review of requirements, models, input–output formats, evaluation metrics, and challenges. *Future Internet*, 15(8):260. 44

Barragán-Montero, A., Javaid, U., Valdés, G., Nguyen, D., Desbordes, P., Macq, B., Willems, S., Vandewinckele, L., Holmström, M., Löfman, F., et al. (2021). Artificial intelligence and machine learning for medical imaging: A technology review. *Physica Medica*, 83:242–256. 3

Ben-David, S., Lu, T., and Pál, D. (2008). Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning. In *Annual Conference Computational Learning Theory*, pages 33–44. 112

Benenson, R. and Ferrari, V. (2022). From colouring-in to pointillism: revisiting semantic segmentation supervision. *arXiv preprint arXiv:2210.14142*. 39, 115

Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.-A., Cetin, I., Lekadir, K., Camara, O., Gonzalez Ballester, M. A., Sanroma, G., Napel, S., Petersen, S., Tziritas, G., Grinias, E., Khened, M., Kollerathu, V. A., Krishnamurthi, G., Rohé, M.-M., Pennec, X., Sermesant, M., Isensee, F., Jäger, P., Maier-Hein, K. H., Full, P. M., Wolf, I., Engelhardt, S., Baumgartner, C. F., Koch, L. M., Wolterink, J. M., Išgum, I., Jang, Y., Hong, Y., Patravali, J., Jain, S., Humbert, O., and Jodoin, P.-M. (2018). Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Transactions on Medical Imaging*, 37(11):2514–2525. 114

Bernardi, F. A., Alves, D., Crepaldi, N., Yamada, D. B., Lima, V. C., and Rijo, R. (2023). Data quality in health research: integrative literature review. *Journal of Medical Internet Research*, 25:e41446. 50

Bezdek, J. C., Ehrlich, R., and Full, W. (1984). Fcm: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3):191–203. 22

Bhagat, P. K. and Choudhary, P. (2018). Image annotation: Then and now. *Image Vis. Comput.*, 80:1–23. 115

Bilic, P., Christ, P., Li, H. B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., Szeskin, A., Jacobs, C., Mamani, G. E. H., Chartrand, G., et al. (2023). The liver tumor segmentation benchmark (lits). *Medical Image Analysis*, 84:102680. 14

Blausen.com staff (2014). Laparoscopy. http://dx.doi.org/10.15347/wjm/2014.010. [Accessed 25-Nov-2024]. 24

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S.,

Chen, A. S., Creel, K. A., Davis, J., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L. E., Goel, K., Goodman, N. D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T. F., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M. S., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S. P., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J. F., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y. H., Ruiz, C., Ryan, J., R'e, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K. P., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M. A., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. (2021). On the opportunities and risks of foundation models. *ArXiv*. 111

Bonfiglio, A., Cannici, M., and Matteucci, M. (2023). Softcut: A fully differentiable relaxed graph cut approach for deep learning image segmentation. In *International Conference on Machine Learning, Optimization, and Data Science*, pages 497–511. Springer. 23

Boutillon, A., Conze, P.-H., Pons, C., Burdin, V., and Borotikar, B. (2022). Generalizable multi-task, multi-domain deep segmentation of sparse pediatric imaging datasets via multi-scale contrastive regularization and multi-joint anatomical priors. *Medical image analysis*, 81:102556. 23

Boykov, Y. and Jolly, M.-P. (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 1, pages 105–112 vol.1. 6, 12, 22, 89, 90

Brady, A. P., Bello, J. A., Derchi, L. E., Fuchsjäger, M., Goergen, S., Krestin, G. P., Lee, E. J., Levin, D. C., Pressacco, J., Rao, V. M., et al. (2021). Radiology in the era of value-based healthcare: a multi-society expert statement from the acr, car, esr, is3r, ranzcr, and rsna. *Canadian Association of Radiologists Journal*, 72(2):208–214. 17

Brett, M., Markiewicz, C. J., Hanke, M., Côté, M.-A., Cipollini, B., McCarthy, P., Jarecka, D., Cheng, C. P., Larson, E., Halchenko, Y. O., Cottaar, M., Ghosh, S., Wassermann, D., Gerhard, S., Lee, G. R., Baratz, Z., Wang, H.-T., Papadopoulos Orfanos, D., Kastman, E., Kaczmarzyk, J., Guidotti, R., Daniel, J., Duek, O., Rokem, A., Scheltienne, M., Madison, C., Sólon, A., Moloney, B., Morency, F. C., Goncalves, M., Markello, R., Riddell, C., Burns, C., Millman, J., Gramfort, A., Leppäkangas, J., van den Bosch, J. J., Vincent, R. D., Braun, H., Subramaniam, K., Van, A., Gorgolewski, K. J., Raamana, Pradeep Reddy andKlug, J., Vos de Wael, R., Nichols, B. N., Baker, E. M., Hayashi, S., Pinsard, B., Haselgrove, C., Hymers, M., Esteban, O., Koudoro, S., Pérez-García, F., Dockès, J., Oosterhof, N. N., Amirbekian, B., Christian, H., Nimmo-Smith, I., Nguyen, L., Suter, P., Reddigari, S., St-Jean, S., Panfilov, E., Garyfallidis, E., Varoquaux, G., Legarreta, J. H., Hahn, K. S., Waller, L., Hinds, O. P., Fauber, B., Dewey, B., Perez, F., Roberts, J., Poline, J.-B., Stutters, J., Jordan, K., Cieslak, M., Moreno, M. E., Hrnčiar, T., Haenel, V., Schwartz, Y., Darwin, B. C., Thirion, B., Gauthier, C., Solovey, I., Gonzalez, Ivan andPalasubramaniam, J., Lecher, J., Leinweber, K., Raktivan, K., Calábková, M., Fischer, P., Gervais, P., Gadde, S., Ballinger, T., Roos, T., Reddam, V. R., and freec84 (2024). nipy/nibabel: 5.2.1. 45

Brown, T. B. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*. 28

Budd, S., Robinson, E. C., and Kainz, B. (2019). A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical image analysis*, 71. 21, 113, 116, 119, 120

Budovec, J. J., Lam, C. A., and Kahn Jr, C. E. (2014). Informatics in radiology: radiology gamuts ontology: differential diagnosis for the semantic web. *Radiographics*, 34(1):254–264. 18

Buntine, W. (2020). Machine learning after the deep learning revolution. *Frontiers of Computer Science*, 14:1–3. 33

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698. 12

Cardenas, C. E., Yang, J., Anderson, B. M., Court, L. E., and Brock, K. B. (2019). Advances in Auto-Segmentation. *Seminars in Radiation Oncology*, 29(3):185–197. 87, 89

Castiglioni, I., Rundo, L., Codari, M., Leo, G. D., Salvatore, C., Interlenghi, M., Gallivanone, F., Cozzi, A., D'Amico, N. C., and Sardanelli, F. (2021). Ai applications to medical images: From machine learning to deep learning. *Physica medica*, 83:9–24. 111

Chaddad, A., Peng, J., Xu, J., and Bouridane, A. (2023). Survey of explainable ai techniques in healthcare. *Sensors*, 23(2):634. 44

Chaisangmongkon, W., Chamveha, I., Promwiset, T., Saiviroonporn, P., and Tongdee, T. (2021). External validation of deep learning algorithms for cardiothoracic ratio measurement. *IEEE Access*, 9:110287–110298. 96

Challen, R., Denny, J., Pitt, M., Gompels, L., Edwards, T., and Tsaneva-Atanasova, K. (2019). Artificial intelligence, bias and clinical safety. *BMJ quality & safety*, 28(3):231–237. 49

Chan, H.-P., Hadjiiski, L. M., and Samala, R. K. (2020). Computer-aided diagnosis in the era of deep learning. *Medical physics*, 47(5):e218–e227. 20

Chauhan, D., Anyanwu, E., Goes, J., Besser, S. A., Anand, S., Madduri, R., Getty, N., Kelle, S., Kawaji, K., Mor-Avi, V., et al. (2022). Comparison of machine learning and deep learning for view identification from cardiac magnetic resonance images. *Clinical imaging*, 82:121–126. 20

Chen, L., Wu, P., Chitta, K., Jaeger, B., Geiger, A., and Li, H. (2024). End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 23

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*. 13

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587. 96

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision*. 99

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. (2021a). Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*. 42

Chen, R. J., Lu, M. Y., Chen, T. Y., Williamson, D. F., and Mahmood, F. (2021b). Synthetic data in machine learning for medicine and healthcare. *Nature Biomedical Engineering*, 5(6):493–497. 20

Chen, X. and Pan, L. (2018). A survey of graph cuts/graph search based medical image segmentation. *IEEE reviews in biomedical engineering*, 11:112–124. 14

Chen, X., Yuan, Y., Zeng, G., and Wang, J. (2021c). Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2613–2622. 134, 135

Chen, X., Zhao, Z., Zhang, Y., Duan, M., Qi, D., and Zhao, H. (2022). Focalclick: Towards practical interactive image segmentation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1290–1299. 106

Cheng, H. K., Tai, Y.-W., and Tang, C.-K. (2021). Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5559–5568. 6

Cho, K. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*. 36

Chollet, F. et al. (2015). Keras. `https://keras.io`. 45

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., and Ronneberger, O. (2016). 3d u-net: learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*, pages 424–432. Springer. 22

Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F., and Ranzuglia, G. (2008). MeshLab: an Open-Source Mesh Processing Tool. In Scarano, V., Chiara, R. D., and Erra, U., editors, *Eurographics Italian Chapter Conference*. The Eurographics Association. 47

Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., et al. (2013). The cancer imaging archive (tcia): maintaining and operating a public information repository. *Journal of digital imaging*, 26:1045–1057. 29

Collins, G. S. and Moons, K. G. (2019). Reporting of artificial intelligence prediction models. *The Lancet*, 393(10181):1577–1579. 49

Collins, T., Pizarro, D., Gasparini, S., Bourdel, N., Chauvet, P., Canis, M., Calvet, L., and Bartoli, A. (2020). Augmented reality guided laparoscopic surgery of the uterus. *IEEE Transactions on Medical Imaging*, 40(1):371–380. 26, 27, 91

Conze, P.-H., Andrade-Miranda, G., Singh, V. K., Jaouen, V., and Visvikis, D. (2023). Current and emerging trends in medical image segmentation with deep learning. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 7(6):545–569. 20, 22, 23

Cooler Master (2024). A brief overview: Ai, ml, dl, and the growing relevance of ai pcs. https://www.coolermaster.com/en-global/guide-and-resources/ai-pc/. [Accessed 26-Nov-2024]. 2

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., and Défossez, A. (2024). Simple and controllable music generation. *Advances in Neural Information Processing Systems*, 36. 42

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223. 6, 39

Criminisi, A., Sharp, T., and Blake, A. (2008). Geos: Geodesic image segmentation. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I 10*, pages 99–112. Springer. 12, 89

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. (2022). Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61. 9

D'Antonoli, T. A., Berger, L. K., Indrakanti, A. K., Vishwanathan, N., Weiß, J., Jung, M., Berkarda, Z., Rau, A., Reisert, M., Küstner, T., et al. (2024). Totalsegmentator mri: Sequence-independent segmentation of 59 anatomical structures in mr images. *arXiv preprint arXiv:2405.19492*. 40, 111

Daye, D., Wiggins, W. F., Lungren, M. P., Alkasab, T., Kottler, N., Allen, B., Roth, C. J., Bizzo, B. C., Durniak, K., Brink, J. A., et al. (2022). Implementation of clinical artificial intelligence in radiology: who decides and how? *Radiology*, 305(3):555–563. 48, 49, 50, 51

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. 39, 97, 124

Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. 36

Dhanachandra, N., Manglem, K., and Chanu, Y. J. (2015). Image segmentation using k-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, 54:764–771. 6, 12

Diaz-Pinto, A., Alle, S., Ihsani, A., Asad, M. H., Nath, V., P'erez-Garc'ia, F., Mehta, P., Li, W., Roth, H. R., Vercauteren, T. K. M., Xu, D., Dogra, P., Ourselin, S., Feng, A., and Cardoso, M. J. (2022). Monai label: A framework for ai-assisted interactive labeling of 3d medical images. *ArXiv*, abs/2203.12362. 46, 90, 117

Dissaux, G., Dissaux, B., El Kabbaj, O., Gujral, D. M., Pradier, O., Salaün, P.-Y., Seizeur, R., Bourhis, D., Salem, D. B., Querellou, S., et al. (2020). Radiotherapy target volume definition in newly diagnosed high grade glioma using 18f-fet pet imaging and multiparametric perfusion mri: a prospective study (imagg). *Radiotherapy and Oncology*, 150:164–171. 67

Dong, S., Wang, P., and Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, 40:100379. 33, 35

Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. 36

Dunn, O. J. (1961). Multiple comparisons among means. *Journal of the American statistical association*, 56(293):52–64. 71, 73

Dutta, S., Lanvin, B., and Wunsch-Vincent, S. (2019). The global innovation index 2019. *Cornell University, INSEAD, & WIPO (Eds.), Global innovation index*, pages 1–400. 50

Dwivedi, R., Dave, D., Naik, H., Singhal, S., Omer, R., Patel, P., Qian, B., Wen, Z., Shah, T., Morgan, G., et al. (2023). Explainable ai (xai): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9):1–33. 44

Electric, G. (2022). MR image reconstruction with AIR™ Recon DL — gehealthcare.com. https://www.gehealthcare.com/products/magnetic-resonance-imaging/air-recon-dl. [Accessed 14-08-2024]. 19

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136. 40, 114

Fan, X., Qiao, X., Wang, Z., Jiang, L., Liu, Y., and Sun, Q. (2022). Artificial intelligence-based ct imaging on diagnosis of patients with lumbar disc herniation by scalpel treatment. *Computational Intelligence and Neuroscience*, 2022(1):3688630. 7

Fix, E. (1985). *Discriminatory analysis: nonparametric discrimination, consistency properties*, volume 1. USAF school of Aviation Medicine. 22

Foroudi, F., Haworth, A., Pangehel, A., Wong, J., Roxby, P., Duchesne, G., Williams, S., and Tai, K. (2009). Inter-observer variability of clinical target volume delineation for bladder cancer using ct and cone beam ct. *Journal of medical imaging and radiation oncology*, 53(1):100–106. 66

French, G., Laine, S., Aila, T., Mackiewicz, M., and Finlayson, G. (2019). Semi-supervised semantic segmentation needs strong, varied perturbations. *arXiv preprint arXiv:1906.01916*. 134

Fujifilm (2024). Synapse 3D. https://synapse-emea.fujifilm.com/synapse-3d.html. [Accessed 23-Sep-2024]. 47, 116

Futrega, M., Milesi, A., Marcinkiewicz, M., and Ribalta, P. (2021). Optimized u-net for brain tumor segmentation. *ArXiv*, abs/2110.03352. 89

Ganaye, P.-A., Sdika, M., Triggs, B., and Benoit-Cattin, H. (2019). Removing segmentation inconsistencies with semi-supervised non-adjacency constraint. *Medical image analysis*, 58:101551. 14

Gao, L., Zhang, Y., Zou, F., Shao, J., and Lai, J. (2020). Unsupervised urban scene segmentation via domain adaptation. *Neurocomputing*, 406:295–301. 7

Garg, R., Bg, V. K., Carneiro, G., and Reid, I. (2016). Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pages 740–756. Springer. 38

Ghiasi, G., Lin, T.-Y., Pang, R., and Le, Q. V. (2019). Nas-fpn: Learning scalable feature pyramid architecture for object detection. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7029–7038. 89

Giattino, C., Mathieu, E., Samborska, V., and Roser, M. (2023). Data page: Domain of notable artificial intelligence systems, by year of publication. https://ourworldindata.org/grapher/domain-notable-artificial-intelligence-systems. Data adapted from Epoch. [Accessed 26-Nov-2024]. 7

Gibson, E., Li, W., Sudre, C., Fidon, L., Shakir, D. I., Wang, G., Eaton-Rosen, Z., Gray, R., Doel, T., Hu, Y., et al. (2018). Niftynet: a deep-learning platform for medical imaging. *Computer methods and programs in biomedicine*, 158:113–122. 22

Goch, C. J., Metzger, J., and Nolden, M. (2017). Abstract: Medical research data management using mitk and xnat - connecting medical image software and data management systems in a research context. In *Bildverarbeitung für die Medizin*. 47, 91, 116

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org. 36

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27. 36

Grady, L. (2006a). Random walks for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 28(11):1768–1783. 12

Grady, L. J. (2006b). Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1768–1783. 89, 115

Gu, B., Ge, R., Chen, Y., Luo, L., and Coatrieux, G. (2020). Automatic and robust object detection in x-ray baggage inspection using deep convolutional neural networks. *IEEE Transactions on Industrial Electronics*, 68(10):10248–10257. 7

Gui, J., Chen, T., Zhang, J., Cao, Q., Sun, Z., Luo, H., and Tao, D. (2024). A survey on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 37

Guo, M.-H., Xu, T.-X., Liu, J.-J., Liu, Z.-N., Jiang, P.-T., Mu, T.-J., Zhang, S.-H., Martin, R. R., Cheng, M.-M., and Hu, S.-M. (2022). Attention mechanisms in computer vision: A survey. *Computational visual media*, 8(3):331–368. 37

Guo, Y., Gao, Y., and Shen, D. (2016). Deformable mr prostate segmentation via deep feature learning and sparse patch matching. *IEEE Transactions on Medical Imaging*, 35(4):1077–1089. 89

Gupta, A., Dollar, P., and Girshick, R. (2019). LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 40

Hadsell, R., Rao, D., Rusu, A. A., and Pascanu, R. (2020). Embracing change: Continual learning in deep neural networks. *Trends in Cognitive Sciences*, 24:1028–1040. 115, 116, 118

Halevy, A., Norvig, P., and Pereira, F. (2009). The unreasonable effectiveness of data. *IEEE intelligent systems*, 24(2):8–12. 29

Han, J., Pei, J., and Tong, H. (2022). *Data mining: concepts and techniques*. Morgan kaufmann. 3, 4

Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., and Malik, J. (2011). Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. 114

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. (2020). Array programming with numpy. *Nature*, 585(7825):357–362. 45

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35:18–31. 89

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916. 13

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. 35, 89, 96, 124, 134

Heim, E., Ross, T., Seitel, A., März, K., Stieltjes, B., Eisenmann, M., Lebert, J., Metzger, J., Sommer, G., Sauter, A. W., Schwartz, F. R., Termer, A., Wagner, F., Kenngott, H., and Maier-Hein, L. (2018). Large-scale medical image annotation with crowd-powered algorithms. *Journal of Medical Imaging*, 5. 115

Hering, A., Hansen, L., Mok, T. C., Chung, A. C., Siebert, H., Häger, S., Lange, A., Kuckertz, S., Heldmann, S., Shao, W., et al. (2022). Learn2reg: comprehensive multi-task medical image registration challenge, dataset and evaluation in the era of deep learning. *IEEE Transactions on Medical Imaging*, 42(3):697–712. 20

Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*. 36

Hofer, I. S., Burns, M., Kendale, S., and Wanderer, J. P. (2020). Realistically integrating machine learning into clinical practice: a road map of opportunities, challenges, and a potential future. *Anesthesia & Analgesia*, 130(5):1115–1118. 48

Holdsworth, C. H., Badawi, R. D., Manola, J. B., Kijewski, M. F., Israel, D. A., Demetri, G. D., and Van den Abbeele, A. D. (2007). Ct and pet: early prognostic indicators of response to imatinib

mesylate in patients with gastrointestinal stromal tumor. *American Journal of Roentgenology*, 189(6):W324–W330. 16

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95. 45

Hussain, M. (2024). Classical-image-segmentation-on-microorganisms. https://github.com/SYED-M-HUSSAIN/Classical-Image-Segmentation-On-Microorganisms. [Accessed 27-Nov-2024]. 13

Iakubovskii, P. (2019). Segmentation models pytorch. https://github.com/qubvel/segmentation_models.pytorch. 45

Iman, M., Arabnia, H. R., and Rasheed, K. (2023). A review of deep transfer learning and recent advancements. *Technologies*, 11(2):40. 20

Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., and Maier-Hein, K. H. (2021). nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211. 22

Isensee, F., Petersen, J., Klein, A., Zimmerer, D., Jaeger, P. F., Kohl, S. A. A., Wasserthal, J., Koehler, G., Norajitra, T., Wirkert, S. J., and Maier-Hein, K. (2018). nnu-net: Self-adapting framework for u-net-based medical image segmentation. *ArXiv*, abs/1809.10486. 106

Ishwarappa, A. J. (2021). Big data based stock trend prediction using deep cnn with reinforcement-lstm model. *International Journal of System Assurance Engineering and Management*, pages 1–11. 7

Islam, M. Z., Islam, M. M., and Asraf, A. (2020). A combined deep cnn-lstm network for the detection of novel coronavirus (covid-19) using x-ray images. *Informatics in medicine unlocked*, 20:100412. 7

Jabeen, S., Li, X., Amin, M. S., Bourahla, O., Li, S., and Jabbar, A. (2023). A review on methods and applications in multimodal deep learning. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(2s):1–41. 38, 39

Jaeger, P. F., Kohl, S. A., Bickelhaupt, S., Isensee, F., Kuder, T. A., Schlemmer, H.-P., and Maier-Hein, K. H. (2020). Retina u-net: Embarrassingly simple exploitation of segmentation supervision for medical object detection. In *Machine Learning for Health Workshop*, pages 171–183. PMLR. 20

Jahanifar, M., Tajeddin, N. Z., Koohbanani, N. A., and Rajpoot, N. M. (2021). Robust interactive semantic segmentation of pathology images with minimal user input. *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 674–683. 89, 90

Jang, W.-D. and Kim, C.-S. (2019). Interactive image segmentation via backpropagating refinement scheme. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5292–5301. 90, 99, 106

Janiesch, C., Zschech, P., and Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3):685–695. 2

JASP Team (2024). JASP (Version 0.18.3)[Computer software]. 71

Jette, M. A. and Wickberg, T. (2023). Architecture of the slurm workload manager. In *Workshop on Job Scheduling Strategies for Parallel Processing*, pages 3–23. Springer. 46

Jha, S. and Topol, E. J. (2016). Adapting to artificial intelligence: radiologists and pathologists as information specialists. *Jama*, 316(22):2353–2354. 50

Ji, Y., Bai, H., Yang, J., Ge, C., Zhu, Y., Zhang, R., Li, Z., Zhang, L., Ma, W., Wan, X., and Luo, P. (2022). Amos: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation. *ArXiv*, abs/2206.08023. 115

Johnson, J. M. and Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of big data*, 6(1):1–54. 8

Johnson, M. and Shotton, J. (2010). Semantic texton forests. In *Computer Vision: Detection, Recognition and Reconstruction*, pages 173–203. Springer. 12

Jones, J. (2024). Normal abdominal x-ray example. https://radiopaedia.org/cases/34067. [Accessed 19-Dec-2024]. 16

Joskowicz, L., Cohen, D., Caplan, N., and Sosna, J. (2019). Inter-observer variability of manual contour delineation of structures in ct. *European radiology*, 29:1391–1399. 67

Juergensen, L., Rischen, R., Toennemann, M., Gosheger, G., Gehweiler, D., and Schulze, M. (2024). Accuracy of pelvic bone segmentation for 3d printing: a study of segmentation accuracy based on anatomic landmarks to evaluate the influence of the observer. *3D Printing in Medicine*, 10(1):33. 67

Jungo, A., Meier, R., Ermis, E., Blatti-Moreno, M., Herrmann, E., Wiest, R., and Reyes, M. (2018). On the effect of inter-observer variability for a reliable estimation of uncertainty of medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I*, pages 682–690. Springer. 66

Kalantar, R., Lin, G., Winfield, J. M., Messiou, C., Lalondrelle, S., Blackledge, M. D., and Koh, D.-M. (2021). Automatic segmentation of pelvic cancers using deep learning: State-of-the-art approaches and challenges. *Diagnostics*, 11(11):1964. 28

Karalis, V. D. (2024). The integration of artificial intelligence into clinical practice. *Applied Biosciences*, 3(1):14–44. 50

Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331. 6

Kato, S. and Hotta, K. (2024). Adaptive t-vmf dice loss: An effective expansion of dice loss for medical image segmentation. *Computers in Biology and Medicine*, 168:107695. 135

Kaur, R. and Singh, S. (2023). A comprehensive review of object detection with deep learning. *Digital Signal Processing*, 132:103812. 4

Kazerouni, A., Aghdam, E. K., Heidari, M., Azad, R., Fayyaz, M., Hacihaliloglu, I., and Merhof, D. (2023). Diffusion models in medical imaging: A comprehensive survey. *Medical Image Analysis*, 88:102846. 20

Ke, Z., Qiu, D., Li, K., Yan, Q., and Lau, R. W. (2020). Guided collaborative training for pixel-wise semi-supervised learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 429–445. Springer. 134

Keshwani, D., Kitamura, Y., Ihara, S., Iizuka, S., and Simo-Serra, E. (2020). Topnet: Topology preserving metric learning for vessel tree reconstruction and labelling. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part VI 23*, pages 14–23. Springer. 14

Khalifa, M. and Albadawy, M. (2024). Ai in diagnostic imaging: Revolutionising accuracy and efficiency. *Computer Methods and Programs in Biomedicine Update*, page 100146. 19

Kiela, D., Bartolo, M., Nie, Y., Kaushik, D., Geiger, A., Wu, Z., Vidgen, B., Prasad, G., Singh, A., Ringshia, P., Ma, Z., Thrush, T., Riedel, S., Waseem, Z., Stenetorp, P., Jia, R., Bansal, M., Potts, C., and Williams, A. (2021). Dynabench: Rethinking benchmarking in NLP. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics. 2

Kiela, D., Thrush, T., Ethayarajh, K., and Singh, A. (2023). Plotting progress in ai. *Contextual AI Blog*. https://contextual.ai/blog/plotting-progress. 2

Kikinis, R., Pieper, S. D., and Vosburgh, K. G. (2013). 3d slicer: a platform for subject-specific image analysis, visualization, and clinical support. In *Intraoperative imaging and image-guided therapy*, pages 277–289. Springer. 47, 91, 116

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026. 6, 111, 118

Kline, T. L., Korfiatis, P., Edwards, M. E., Blais, J. D., Czerwiec, F. S., Harris, P. C., King, B. F., Torres, V. E., and Erickson, B. J. (2017). Performance of an artificial multi-observer deep neural network for fully automated segmentation of polycystic kidneys. *Journal of Digital Imaging*, 30(4):442–448. 89

Koohbanani, N. A., Jahanifar, M., Tajadin, N. Z., and Rajpoot, N. M. (2020). Nuclick: A deep learning framework for interactive segmentation of microscopy images. *Medical image analysis*, 65. 90, 99

Krenn, M., Buffoni, L., Coutinho, B., Eppel, S., Foster, J. G., Gritsevskiy, A., Lee, H., Lu, Y., Moutinho, J. P., Sanjabi, N., et al. (2022). Predicting the future of ai with ai: High-quality link prediction in an exponentially growing knowledge network. *arXiv preprint arXiv:2210.00881*. 33

Kruskal, W. H. and Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621. 71

Kusakunniran, W., Saiviroonporn, P., Siriapisith, T., Tongdee, T., Uraiverotchanakorn, A., Leesakul, S., Thongnarintr, P., Kuama, A., and Yodprom, P. (2023). Automatic measurement of cardiothoracic ratio in chest x-ray images with progan-generated dataset. *Applied Computing and Informatics*. 96

Langlotz, C. et al. (2019). A roadmap for foundational research on artificial intelligence in medical imaging: From the 2018 nih/rsna/acr/the academy workshop. *Radiology*, 291 3:781–791. 115

Lecart, M. (2024). *Analyse de corrélation inter-observateur concernant la segmentation d'utérus myomateux en IRM. Médecine humaine et pathologie*. PhD thesis, Université Clermont Auvergne". Accessed: 2024-8-13. 18, 30

Le'Clerc Arrastia, J., Heilenkötter, N., Otero Baguer, D., Hauberg-Lotte, L., Boskamp, T., Hetzer, S., Duschner, N., Schaller, J., and Maass, P. (2021). Deeply Supervised UNet for Semantic Segmentation to Assist Dermatopathological Assessment of Basal Cell Carcinoma. *J Imaging*, 7(4). 98

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551. 13

Lee, K., Lee, K., Shin, J., and Lee, H. (2019). Overcoming catastrophic forgetting with unlabeled data in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 312–321. 8

Lee, Y., Yoon, S., Paek, M., Han, D., Choi, M. H., and Park, S. H. (2024). Advanced mri techniques in abdominal imaging. *Abdominal Radiology*, pages 1–22. 27

Lei, L., Yang, Q., Yang, L., Shen, T., Wang, R., and Fu, C. (2024). Deep learning implementation of image segmentation in agricultural applications: a comprehensive review. *Artificial Intelligence Review*, 57(6):149. 23

Leibetseder, A., Schoeffmann, K., Keckstein, J., and Keckstein, S. (2022). Endometriosis detection and localization in laparoscopic gynecology. *Multimedia Tools and Applications*, 81(5):6191–6215. 20

Lepcha, D. C., Goyal, B., Dogra, A., Sharma, K. P., and Gupta, D. N. (2023). A deep journey into image enhancement: A survey of current and emerging trends. *Information Fusion*, 93:36–76. 14, 19

Li, H., Zhao, R., and Wang, X. (2014). Highly efficient forward and backward propagation of convolutional neural networks for pixelwise classification. *arXiv preprint arXiv:1412.4526*. 13

Li, J., Zhou, Z., Yang, J., Pepe, A., Gsaxner, C., Luijten, G., Qu, C., Zhang, T., Chen, X., Li, W., et al. (2023). Medshapenet–a large-scale dataset of 3d medical shapes for computer vision. *arXiv preprint arXiv:2308.16139*. 41

Li, W., Qu, C., Chen, X., Bassi, P. R., Shi, Y., Lai, Y., Yu, Q., Xue, H., Chen, Y., Lin, X., et al. (2024). Abdomenatlas: A large-scale, detailed-annotated, & multi-center dataset for efficient transfer learning and open algorithmic benchmarking. *Medical Image Analysis*, 97:103285. 41, 115

Li, X., Zhong, Z., Wu, J., Yang, Y., Lin, Z., and Liu, H. (2019). Expectation-maximization attention networks for semantic segmentation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9166–9175. 89

Liao, X., Li, W., Xu, Q., Wang, X., Jin, B., Zhang, X., Zhang, Y., and Wang, Y. (2020). Iteratively-refined interactive 3d medical image segmentation with multi-agent reinforcement learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9391–9399. 89, 90, 93, 106, 117

Lim, K., Small Jr, W., Portelance, L., Creutzberg, C., Jürgenliemk-Schulz, I. M., Mundt, A., Mell, L. K., Mayr, N., Viswanathan, A., Jhingran, A., et al. (2011). Consensus guidelines for delineation of clinical target volume for intensity-modulated pelvic radiotherapy for the definitive treatment of cervix cancer. *International Journal of Radiation Oncology* Biology* Physics*, 79(2):348–355. 67

Lin, T.-Y., Goyal, P., Girshick, R. B., He, K., and Dollár, P. (2017). Focal loss for dense object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007. 98, 124

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014a). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer. 39, 40

Lin, T.-Y., Maire, M., Belongie, S. J., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014b). Microsoft coco: Common objects in context. In *European Conference on Computer Vision*. 115

Lin, Z., Duan, Z.-P., Zhang, Z., Guo, C.-L., and Cheng, M.-M. (2022). Focuscut: Diving into a focus view in interactive segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2637–2646. 106

Lindner, C., Thiagarajah, S., Wilkinson, J. M., Wallis, G. A., Cootes, T. F., arcOGEN Consortium, et al. (2013). Fully automatic segmentation of the proximal femur using random forest regression voting. *IEEE transactions on medical imaging*, 32(8):1462–1472. 12

Liu, X., Han, C., Wang, H., Wu, J., Cui, Y., Zhang, X., and Wang, X. (2021). Fully automated pelvic bone segmentation in multiparameteric mri using a 3d convolutional neural network. *Insights into Imaging*, 12:1–13. 28

Liu, X., Rivera, S. C., Moher, D., Calvert, M. J., Denniston, A. K., Ashrafian, H., Beam, A. L., Chan, A.-W., Collins, G. S., Deeks, A. D. J., et al. (2020). Reporting guidelines for clinical trial reports for interventions involving artificial intelligence: the consort-ai extension. *The Lancet Digital Health*, 2(10):e537–e548. 49

Liu, Y., Han, T., Ma, S., Zhang, J., Yang, Y., Tian, J., He, H., Li, A., He, M., Liu, Z., et al. (2023). Summary of chatgpt-related research and perspective towards the future of large language models. *Meta-Radiology*, page 100017. 9, 36, 42

Liu, Z., Wang, J., Gong, S., Tao, D., and Lu, H. (2019). Deep reinforcement active learning for human-in-the-loop person re-identification. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6121–6130. 116

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440. 13

Lu, X., Xie, Q., Zha, Y., and Wang, D. (2018). Fully automatic liver segmentation combining multi-dimensional graph cut with shape information in 3d ct images. *Scientific reports*, 8(1):10700. 14

Luebke, D. (2008). Cuda: Scalable parallel programming for high-performance scientific computing. In *2008 5th IEEE international symposium on biomedical imaging: from nano to macro*, pages 836–838. IEEE. 45

Lundberg, S. (2017). A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*. 44

Luo, X., Hu, M., Song, T., Wang, G., and Zhang, S. (2022). Semi-supervised medical image segmentation via cross teaching between cnn and transformer. In *International conference on medical imaging with deep learning*, pages 820–833. PMLR. 135

Luo, X., Wang, G., Song, T., Zhang, J., Aertsen, M., Deprest, J., Ourselin, S., Vercauteren, T., and Zhang, S. (2021). Mideepseg: Minimally interactive segmentation of unseen objects from medical images using deep learning. *Medical image analysis*, 72:102102. 23

Maccioni, F., Busato, L., Valenti, A., Cardaccio, S., Longhi, A., and Catalano, C. (2023). Magnetic resonance imaging of the gastrointestinal tract: current role, recent advancements and future prospectives. *Diagnostics*, 13(14):2410. 27

Majee, A. (2024). Deepops & slurm: Your gpu cluster guide. *arXiv preprint arXiv:2405.00030*. 47

Maslej, N., Fattorini, L., Perrault, R., Parli, V., Reuel, A., Brynjolfsson, E., Etchemendy, J., Ligett, K., Lyons, T., Manyika, J., Niebles, J. C., Shoham, Y., Wald, R., and Clark, J. (2024). Artificial intelligence index report 2024. 33, 40, 41, 42, 43

McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*. 64

McKesson Corporation (2024). Mckesson - medical supplies, pharmaceuticals healthcare solutions. https://www.mckesson.com/. [Accessed 22-10-2024]. 58

Meta (2024). Building Meta's GenAI Infrastructure. https://engineering.fb.com/2024/03/12/data-center-engineering/building-metas-genai-infrastructure/. [Accessed 23-09-2024]. 46

Midjourney, I. (2022). Midkourney. 42

Mikhailov, I. and Bartoli, A. (2024a). Système et procédé combiné de sélection, d'annotation et d'entraînement via un modèle d'apprentissage automatique partagé. https://data.inpi.fr/brevets/FR3146529?q. [Accessed 18-Dec-2024]. xii

Mikhailov, I. and Bartoli, A. (2024b). Système et procédé de segmentation d'image semi-automatique par apprentissage à boucle d'interaction utilisateur et procédé d'entraînement associé. https://data.inpi.fr/brevets/FR3139651?q. [Accessed 18-Dec-2024]. xii

Mikhailov, I., Chauveau, B., Bourdel, N., and Bartoli, A. (2022). A deep learning-based interactive medical image segmentation framework. In Wu, S., Shabestari, B., and Xing, L., editors, *Applications of Medical Artificial Intelligence*, pages 98–107, Cham. Springer Nature Switzerland. 31

Mikhailov, I., Chauveau, B., Bourdel, N., and Bartoli, A. (2023). Sharing is caring: Concurrent interactive segmentation and model training using a joint model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2432–2441. 31, 121, 129

Mikhailov, I., Chauveau, B., Bourdel, N., and Bartoli, A. (2024). A deep learning-based interactive medical image segmentation framework with sequential memory. *Computer Methods and Programs in Biomedicine*, 245. 31, 117, 118, 121, 122

Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., and Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542. 5, 89, 115

Minig, L. (2024). Minimally invasive laparoscopic surgery for treatment of gynecological disease. https://drlucasminig.com/en/treatment-by-mini-invasive-laparoscopic-surgery/. [Accessed 25-Nov-2024]. 24

Monnet, E. and Twedt, D. C. (2003). Laparoscopy. *Veterinary Clinics: Small Animal Practice*, 33(5):1147–1163. 25

Montagne, S., Hamzaoui, D., Allera, A., Ezziane, M., Luzurier, A., Quint, R., Kalai, M., Ayache, N., Delingette, H., and Renard-Penna, R. (2021). Challenge of prostate mri segmentation on t2-weighted images: inter-observer variability and impact of prostate morphology. *Insights into imaging*, 12(1):71. 66

Mortensen, E. N. and Barrett, W. A. (1995). Intelligent scissors for image composition. *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 113, 115

Mosqueira-Rey, E., Hernández-Pereira, E., Alonso-Ríos, D., Bobes-Bascarán, J., and Fernández-Leal, Á. (2023). Human-in-the-loop machine learning: a state of the art. *Artificial Intelligence Review*, 56(4):3005–3054. 11, 12

Muhammad, K., Hussain, T., Ullah, H., Del Ser, J., Rezaei, M., Kumar, N., Hijji, M., Bellavista, P., and de Albuquerque, V. H. C. (2022). Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):22694–22715. 7

Murtaza, H., Ahmed, M., Khan, N. F., Murtaza, G., Zafar, S., and Bano, A. (2023). Synthetic data generation: State of the art in health care domain. *Computer Science Review*, 48:100546. 20

Najjar, R. (2023). Redefining radiology: a review of artificial intelligence integration in medical imaging. *Diagnostics*, 13(17):2760. 7, 19

Najman, L. and Schmitt, M. (1994). Watershed of a continuous function. *Signal processing*, 38(1):99–112. 6, 12

Ng, A. (2016). How scale is enabling deep learning. https://youtu.be/LcfLo7YP8O4. Accessed 13-Jan-2025. 4

Nock, R. and Nielsen, F. (2004). Statistical region merging. *IEEE Transactions on pattern analysis and machine intelligence*, 26(11):1452–1458. 6, 12

Nosrati, M. S. and Hamarneh, G. (2016). Incorporating prior knowledge in medical image segmentation: a survey. *arXiv preprint arXiv:1607.01092*. 14

Nwanosike, E. M., Conway, B. R., Merchant, H. A., and Hasan, S. S. (2022). Potential applications and performance of machine learning techniques and algorithms in clinical practice: a systematic review. *International journal of medical informatics*, 159:104679. 48

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., et al. (2018). Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*. 14

Olatunji, I. E., Rauch, J., Katzensteiner, M., and Khosla, M. (2021). A review of anonymization for healthcare data. *Big Data*. PMID: 35271377. 18

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., and Walsh, J. (2020). Deep learning vs. traditional computer vision. In Arai, K. and Kapoor, S., editors, *Advances in Computer Vision*, pages 128–144, Cham. Springer International Publishing. 87

Omouri, A., Rapacchi, S., Duclos, J., Niddam, R., Bellemare, M.-E., and Pirró, N. (2024). 3d observation of pelvic organs with dynamic mri segmentation: A bridge toward patient-specific models. *International Urogynecology Journal*, pages 1–9. 28

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66. 22, 113

Otsu, N. et al. (1975). A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27. 6, 12

Ouali, Y., Hudelot, C., and Tami, M. (2020). Semi-supervised semantic segmentation with cross-consistency training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12674–12684. 134

Özbey, M., Dalmaz, O., Dar, S. U., Bedel, H. A., Özturk, Ş., Güngör, A., and Çukur, T. (2023). Unsupervised medical image translation with adversarial diffusion models. *IEEE Transactions on Medical Imaging*. 19

Pan, H., Chen, M., Bai, W., Li, B., Zhao, X., Zhang, M., Zhang, D., Li, Y., Wang, H., Geng, H., et al. (2024). Large-scale uterine myoma mri dataset covering all figo types with pixel-level annotations. *Scientific Data*, 11(1):410. 28, 41, 53, 139

Paoletti, M. E., Haut, J. M., Plaza, J., and Plaza, A. (2019). Deep learning classifiers for hyperspectral imaging: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:279–317. 7

Parasher, A. and Mohan, V. (2021). Postpartum fatal fulminant hepatic failure presenting with persistent hypoglycemia due to acute fatty liver of pregnancy. *Journal of Medical Sciences*, 41(2):99–104. 16

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32. 45

Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572. 4

Pinter, C., Lasso, A., Wang, A., Jaffray, D., and Fichtinger, G. (2012). Slicerrt – radiation therapy research toolkit for 3d slicer. *Med. Phys.*, 39(10):6332–6338. 68

Plath, N., Toussaint, M., and Nakajima, S. (2009). Multi-class image segmentation using conditional random fields and global classification. In *Proceedings of the 26th annual international conference on machine learning*, pages 817–824. 6

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. (2023). Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*. 42

Proscia, N., Jaffe, T. A., Neville, A. M., Wang, C. L., Dale, B. M., and Merkle, E. M. (2010). Mri of the pelvis in women: 3d versus 2d t2-weighted technique. *American Journal of Roentgenology*, 195(1):254–259. 53

Pulido, J. V., Guleria, S., Ehsan, L., Fasullo, M., Lippman, R., Mutha, P., Shah, T., Syed, S., and Brown, D. E. (2020). Semi-supervised classification of noisy, gigapixel histology images. In *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 563–568. IEEE. 112

Puttagunta, M. K., Ravi, S., and Nelson Kennedy Babu, C. (2023). Adversarial examples: attacks and defences on medical deep learning systems. *Multimedia Tools and Applications*, 82(22):33773–33809. 49

Qureshi, I., Yan, J., Abbas, Q., Shaheed, K., Riaz, A. B., Wahid, A., Khan, M. W. J., and Szczuko, P. (2023). Medical image segmentation using deep semantic-based methods: A review of techniques, applications and emerging trends. *Information Fusion*, 90:316–352. 20

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR. 43

Ramadan, H., Lachqar, C., and Tairi, H. (2020). A survey of recent interactive image segmentation methods. *Computational Visual Media*, 6:355 – 384. 87

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr. 43

Raschka, S., Patterson, J., and Nolet, C. (2020). Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *Information*, 11(4):193. 45, 46

Ravi, N., Gabeur, V., Hu, Y.-T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K. V., Carion, N., Wu, C.-Y., Girshick, R., Dollár, P., and Feichtenhofer, C. (2024). Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*. 6, 13, 14, 139

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Chen, X., and Wang, X. (2020). A survey of deep active learning. *ACM Computing Surveys (CSUR)*, 54:1 – 40. 38, 113, 116, 120

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144. 44

Robinson, T. and Stiegmann, G. (2004). Minimally invasive surgery. *Endoscopy*, 36(01):48–51. 24

Robinson, T. and Stiegmann, G. (2007). Minimally invasive surgery. *Endoscopy*, 39(01):21–23. 24

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer. 13, 22, 35, 89

Rosa, C., Pizzi, A. D., Augurio, A., Caravatta, L., Di Tommaso, M., Mincuzzi, E., Cinalli, S., Basilico, R., Porreca, A., Di Nicola, M., et al. (2020). Volume delineation in cervical cancer with t2 and diffusion-weighted mri: agreement on volumes between observers. *in vivo*, 34(4):1981–1986. 67

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408. 1

Rother, C., Kolmogorov, V., and Blake, A. (2004). "grabcut": interactive foreground extraction using iterated graph cuts. *ACM SIGGRAPH 2004 Papers*. 115

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536. 89

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2014). Imagenet large scale visual recognition challenge. arxiv e-prints, page. *arXiv preprint arXiv:1409.0575*, 2(4). 46

Sadri, A. R., Janowczyk, A., Zhou, R., Verma, R., Beig, N., Antunes, J., Madabhushi, A., Tiwari, P., and Viswanath, S. E. (2020). Mrqy—an open-source tool for quality control of mr imaging data. *Medical physics*, 47(12):6029–6038. 45, 64, 65

Sakala, M. D., Shampain, K. L., and Wasnik, A. P. (2020). Advances in mr imaging of the female pelvis. *Magnetic Resonance Imaging Clinics*, 28(3):415–431. 27, 53

Sakinis, T., Milletari, F., Roth, H. R., Korfiatis, P., Kostandy, P. M., Philbrick, K. A., Akkus, Z., Xu, Z., Xu, D., and Erickson, B. J. (2019). Interactive segmentation of medical images through fully convolutional neural networks. *ArXiv*, abs/1903.08205. 89, 90, 106

Salehin, I., Islam, M. S., Saha, P., Noman, S., Tuni, A., Hasan, M. M., and Baten, M. A. (2024). Automl: A systematic review on automated machine learning with neural architecture search. *Journal of Information and Intelligence*, 2(1):52–81. 38

Salvi, M., Loh, H. W., Seoni, S., Barua, P. D., García, S., Molinari, F., and Acharya, U. R. (2023). Multi-modality approaches for medical support systems: A systematic review of the last decade. *Information Fusion*, page 102134. 43

Sarker, I. H. (2021a). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, 2(6):420. 7, 34, 35, 36

Sarker, I. H. (2021b). Machine learning: Algorithms, real-world applications and research directions. *SN Comput. Sci.*, 2(3). 3

Sarker, I. H., Abushark, Y. B., Alsolami, F., and Khan, A. I. (2020a). Intrudtree: a machine learning based cyber security intrusion detection model. *Symmetry*, 12(5):754. 4

Sarker, I. H., Abushark, Y. B., and Khan, A. I. (2020b). Contextpca: Predicting context-aware smartphone apps usage based on machine learning techniques. *Symmetry*, 12(4):499. 4

Schmarje, L., Santarossa, M., Schroder, S.-M., and Koch, R. (2020). A survey on semi-, self- and unsupervised learning for image classification. *IEEE Access*, 9:82146–82168. 111

Seidel, A., Krattenmacher, N., and Thaller, G. (2020). Dealing with complexity of new phenotypes in modern dairy cattle breeding. *Animal Frontiers*, 10(2):23–28. 3

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626. 44

Sengupta, S., Basak, S., Saikia, P., Paul, S., Tsalavoutis, V., Atiah, F., Ravi, V., and Peters, A. (2020). A review of deep learning with special emphasis on architectures, applications and recent trends. *Knowledge-Based Systems*, 194:105596. 33

Sevakula, R. K., Singh, V., Verma, N. K., Kumar, C., and Cui, Y. (2018). Transfer learning for molecular cancer classification using deep neural networks. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(6):2089–2100. 7

Shamshad, F., Khan, S., Zamir, S. W., Khan, M. H., Hayat, M., Khan, F. S., and Fu, H. (2023). Transformers in medical imaging: A survey. *Medical Image Analysis*, 88:102802. 44

Shan, F., Gao, Y., Wang, J., Shi, W., Shi, N., Han, M., Xue, Z., Shen, D., and Shi, Y. (2020). Lung infection quantification of covid-19 in ct images with deep learning. *ArXiv*. 89

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423. 120

Sharma, N. and Aggarwal, L. M. (2010). Automated medical image segmentation techniques. *Journal of medical physics*, 35(1):3–14. 22

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306. 13, 124

Shi, Y. (2016). Understanding lstm and its diagrams. https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714. [Accessed 27-Nov-2024]. 37

Shinde, P. P. and Shah, S. (2018). A review of machine learning and deep learning applications. In *2018 Fourth international conference on computing communication control and automation (ICCUBEA)*, pages 1–6. IEEE. 3

Shumailov, I., Shumaylov, Z., Zhao, Y., Gal, Y., Papernot, N., and Anderson, R. (2023). The curse of recursion: Training on generated data makes models forget. *arXiv preprint arXiv:2305.17493*. 44

Shvets, A. A., Iglovikov, V. I., Rakhlin, A., and Kalinin, A. A. (2018). Angiodysplasia detection and localization using deep convolutional neural networks. *bioRxiv*. 96

Siddique, N., Paheding, S., Elkin, C. P., and Devabhaktuni, V. (2021). U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9:82031–82057. 89

Siddiquee, M. M. R. and Myronenko, A. (2021). Redundancy reduction in semantic segmentation of 3d brain tumor mris. *ArXiv*, abs/2111.00742. 89

Sim, J. Z. T., Fong, Q. W., Huang, W., and Tan, C. H. (2023). Machine learning in medicine: what clinicians should know. *Singapore medical journal*, 64(2):91–97. 49, 50

Simpson, A. L., Antonelli, M., Bakas, S., Bilello, M., Farahani, K., van Ginneken, B., Kopp-Schneider, A., Landman, B. A., Litjens, G. J. S., Menze, B. H., Ronneberger, O., Summers, R. M., Bilic, P., Christ, P. F., Do, R. K. G., Gollub, M. J., Golia-Pernicka, J., Heckers, S., Jarnagin, W. R., McHugo, M., Napel, S., Vorontsov, E., Maier-Hein, L., and Cardoso, M. J. (2019). A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *ArXiv*, abs/1902.09063. 89, 101

Singla, A., Sukharevsky, A., Yee, L., Chui, M., and Hall, B. (2024). The state of ai in early 2024: Gen ai adoption spikes and starts to generate value. *McKinsey and Company*. 48

Smith, J. R., Del Priore, G., Coleman, R. L., and Monaghan, J. M. (2018). *An atlas of gynecologic oncology: investigation and surgery*. CRC Press. 25

Sofiiuk, K., Petrov, I. A., Barinova, O., and Konushin, A. (2020). F-brs: Rethinking backpropagating refinement for interactive segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8620–8629. 90, 99, 106

Sofiiuk, K., Petrov, I. A., and Konushin, A. (2021). Reviving iterative training with mask guidance for interactive segmentation. *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3141–3145. 90

Sohn, K., Berthelot, D., Li, C.-L., Zhang, Z., Carlini, N., Cubuk, E. D., Kurakin, A., Zhang, H., and Raffel, C. (2020). Fixmatch: simplifying semi-supervised learning with consistency and confidence. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc. 117

Song, Y., Wang, T., Cai, P., Mondal, S. K., and Sahoo, J. P. (2023). A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Computing Surveys*, 55(13s):1–40. 20, 37

Sounderajah, V., Ashrafian, H., Aggarwal, R., De Fauw, J., Denniston, A. K., Greaves, F., Karthikesalingam, A., King, D., Liu, X., Markar, S. R., et al. (2020). Developing specific reporting guidelines for diagnostic accuracy studies assessing ai interventions: The stard-ai steering group. *Nature medicine*, 26(6):807–808. 49

Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101. 71

Srinath, K. (2017). Python–the fastest growing programming language. *International Research Journal of Engineering and Technology*, 4(12):354–357. 45

Starck, J.-L., Elad, M., and Donoho, D. L. (2005). Image decomposition via the combination of sparse representations and a variational approach. *IEEE transactions on image processing*, 14(10):1570–1582. 6

Su, H., Deng, J., and Fei-Fei, L. (2012). Crowdsourcing annotations for visual object detection. In *Workshops at the twenty-sixth AAAI conference on artificial intelligence*. 10

Suetens, P. (2017). *Fundamentals of medical imaging*. Cambridge university press. 14, 15

Suganyadevi, S., Seethalakshmi, V., and Balasamy, K. (2022). A review on deep learning in medical image analysis. *International Journal of Multimedia Information Retrieval*, 11(1):19–38. 35

Supervisely OU (2024). Supervisely. https://supervise.ly/. [Accessed 23-Sep-2024]. 47, 116

SURGAR (2024). Augmented reality software for minimally invasive surgery. https://surgar-surgery.com/. [Accessed 25-Nov-2024]. 16, 26, 138

Suzuki, S. et al. (1985). Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1):32–46. 12

Szeliski, R. (2022). *Computer vision: algorithms and applications*. Springer Nature. 5

Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J. N., Wu, Z., and Ding, X. (2020). Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. *Medical image analysis*, 63. 6, 9, 20, 113, 115, 116

Talaei Khoei, T., Ould Slimane, H., and Kaabouch, N. (2023). Deep learning: systematic review, models, challenges, and research directions. *Neural Computing and Applications*, 35. 2

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. (2018). A survey on deep transfer learning. In *International Conference on Artificial Neural Networks*. 111

Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., et al. (2023). Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*. 43

Tharwat, A. and Schenck, W. (2023). A survey on active learning: State-of-the-art, practical challenges and research directions. *Mathematics*, 11(4):820. 10, 38

The European Parliament and the Council of the European Union (2024a). Finding endometriosis using machine learning (female). https://findingendometriosis.eu/. [Accessed 31-Oct-2024]. 85

The European Parliament and the Council of the European Union (2024b). Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence and amending various regulations (artificial intelligence act). https://eur-lex.europa.eu/eli/reg/2024/1689/oj. Accessed: 2024-09-25. 50

Thiriveedhi, V. K., Krishnaswamy, D., Clunie, D., Pieper, S., Kikinis, R., and Fedorov, A. (2024). Cloud-based large-scale curation of medical imaging data using ai segmentation. *Research Square*. 41

Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497. 22

Tufail, S., Riggs, H., Tariq, M., and Sarwat, A. I. (2023). Advancements and challenges in machine learning: A comprehensive review of models, libraries, applications, and algorithms. *Electronics*, 12(8):1789. 45

Tustison, N., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P., and Gee, J. C. (2010a). N4itk: Improved n3 bias correction. *IEEE Transactions on Medical Imaging*, 29:1310–1320. 124

Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., and Gee, J. C. (2010b). N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310–1320. 19, 98

Unberath, M., Gao, C., Hu, Y., Judish, M., Taylor, R. H., Armand, M., and Grupp, R. (2021). The impact of machine learning on 2d/3d registration for image-guided interventions: A systematic review and perspective. *Frontiers in Robotics and AI*, 8:716007. 4

Usamentiaga, R., Lema, D. G., Pedrayes, O. D., and Garcia, D. F. (2022). Automated surface defect detection in metals: a comparative review of object detection and semantic segmentation using deep learning. *IEEE Transactions on Industry Applications*, 58(3):4203–4213. 23

Van der Velden, B. H., Kuijf, H. J., Gilhuijs, K. G., and Viergever, M. A. (2022). Explainable artificial intelligence (xai) in deep learning-based medical image analysis. *Medical Image Analysis*, 79:102470. 21

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*. 13, 23, 36

Vettoruzzo, A., Bouguelia, M.-R., Vanschoren, J., Rognvaldsson, T., and Santosh, K. (2024). Advances and challenges in meta-learning: A technical review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 38

Virzì, A., Muller, C. O., Marret, J.-B., Mille, E., Berteloot, L., Grévent, D., Boddaert, N., Gori, P., Sarnacki, S., and Bloch, I. (2020). Comprehensive review of 3d segmentation software tools for mri usable for pelvic surgery planning. *Journal of digital imaging*, 33:99–110. 27, 28

Vrooman, H. A., Cocosco, C. A., Stokking, R., Ikram, M. A., Vernooij, M. W., Breteler, M. M., and Niessen, W. J. (2006). kNN-based multi-spectral MRI brain tissue classification: manual training versus automated atlas-based training. In Reinhardt, J. M. and Pluim, J. P. W., editors, *Medical Imaging 2006: Image Processing*, volume 6144, pages 1142 – 1150. International Society for Optics and Photonics, SPIE. 89, 90

Waldrop, M. M. (2019). What are the limits of deep learning? *Proceedings of the National Academy of Sciences*, 116(4):1074–1077. 5

Wang, G., Li, W., Zuluaga, M. A., Pratt, R., Patel, P. A., Aertsen, M., Doel, T., David, A. L., Deprest, J. A., Ourselin, S., and Vercauteren, T. K. M. (2018). Interactive medical image segmentation using deep learning with image-specific fine tuning. *IEEE Transactions on Medical Imaging*, 37:1562–1573. 89, 90, 92

Wang, G., Zuluaga, M. A., Li, W., Pratt, R., Patel, P. A., Aertsen, M., Doel, T., David, A. L., Deprest, J. A., Ourselin, S., and Vercauteren, T. K. M. (2019a). Deepigeos: A deep interactive geodesic framework for medical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:1559–1572. 89, 92

Wang, H. and Raj, B. (2017). On the origin of deep learning. *arXiv preprint arXiv:1702.07800*. 33

Wang, J., Yu, L.-C., Lai, K. R., and Zhang, X. (2019b). Tree-structured regional cnn-lstm model for dimensional sentiment analysis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:581–591. 7

Wang, L., Zhang, X., Su, H., and Zhu, J. (2024). A comprehensive survey of continual learning: theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 37

Wang, Y., Yao, Q., Kwok, J. T.-Y., and shuan Ni, L. M. (2019c). Generalizing from a few examples: A survey on few-shot learning. *arXiv: Learning*. 111

Wang, Z., Yang, E., Shen, L., and Huang, H. (2023). A comprehensive survey of forgetting in deep learning beyond continual learning. *arXiv preprint arXiv:2307.09218*. 8

Ward, T. M., Fer, D. M., Ban, Y., Rosman, G., Meireles, O. R., and Hashimoto, D. A. (2021). Challenges in surgical video annotation. *Computer Assisted Surgery*, 26(1):58–68. 29

Warfield, S. K., Zou, K. H., and Wells, W. M. (2004). Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, 23(7):903–921. 70

Wasserthal, J., Breit, H.-C., Meyer, M. T., Pradella, M., Hinck, D., Sauter, A. W., Heye, T., Boll, D. T., Cyriac, J., Yang, S., Bach, M., and Segeroth, M. (2023). Totalsegmentator: Robust segmentation of 104 anatomic structures in ct images. *Radiology: Artificial Intelligence*, 5(5). 40, 111

Wen, J., Zhang, Z., Lan, Y., Cui, Z., Cai, J., and Zhang, W. (2023). A survey on federated learning: challenges and applications. *International Journal of Machine Learning and Cybernetics*, 14(2):513–535. 39

Westbrook, C. and Talbot, J. (2018). *MRI in Practice*. John Wiley & Sons. 27, 28

Whang, S. E., Roh, Y., Song, H., and Lee, J.-G. (2023). Data collection and quality challenges in deep learning: A data-centric ai perspective. *The VLDB Journal*, 32(4):791–813. 9

Wiener, N. (2019). *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press. 13

Wightman, R. (2019). Pytorch image models. https://github.com/rwightman/pytorch-image-models. 45

Willemink, M. J., Koszek, W. A., Hardell, C., Wu, J., Fleischmann, D., Harvey, H., Folio, L. R., Summers, R. M., Rubin, D., and Lungren, M. P. (2020a). Preparing medical imaging data for machine learning. *Radiology*. 115

Willemink, M. J., Koszek, W. A., Hardell, C., Wu, J., Fleischmann, D., Harvey, H., Folio, L. R., Summers, R. M., Rubin, D. L., and Lungren, M. P. (2020b). Preparing medical imaging data for machine learning. *Radiology*, 295(1):4–15. 50

Wong, V. W. H., Ferguson, M., Law, K. H., and Lee, Y.-T. T. (2019). An assistive learning workflow on annotating images for object detection. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 1962–1970. 117

Woo, M., Heo, M., Devane, A. M., Lowe, S. C., and Gimbel, R. W. (2020). Retrospective comparison of approaches to evaluating inter-observer variability in ct tumour measurements in an academic health centre. *BMJ open*, 10(11):e040096. 67

Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., and He, L. (2021). A survey of human-in-the-loop for machine learning. *Future Gener. Comput. Syst.*, 135:364–381. 21, 113

Xie, H., Xu, W., Wang, Y. X., Buatti, J., and Wu, X. (2023). gcdlseg: Integrating graph-cut into deep learning for binary semantic segmentation. *arXiv preprint arXiv:2312.04713*. 14

Xie, Q., Hovy, E. H., Luong, M.-T., and Le, Q. V. (2020). Self-training with noisy student improves imagenet classification. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695. 89

Xie, X., Niu, J., Liu, X., Chen, Z., Tang, S., and Yu, S. (2021). A survey on incorporating domain knowledge into deep learning for medical image analysis. *Medical Image Analysis*, 69:101985. 23

Xu, Z., Huo, Y., Park, J., Landman, B., Milkowski, A., Grbic, S., and Zhou, S. (2018). Less is more: Simultaneous view classification and landmark detection for abdominal ultrasound images. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, pages 711–719. Springer. 20

Xue, H., An, Y., Qin, Y., Li, W., Wu, Y., Che, Y., Fang, P., and Zhang, M. (2024). Towards few-shot learning in the open world: A review and beyond. *arXiv preprint arXiv:2408.09722*. 37

Yang, L., Qi, L., Feng, L., Zhang, W., and Shi, Y. (2023a). Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7236–7246. 114, 117, 134, 135

Yang, L., Zhuo, W., Qi, L., Shi, Y., and Gao, Y. (2022). St++: Make self-trainingwork better for semi-supervised semantic segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4258–4267. 114, 117, 134

Yang, X., Song, Z., King, I., and Xu, Z. (2023b). A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):8934–8954. 112, 116

Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., and Chen, E. (2023). A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*. 43

Yoo, D. and Kweon, I. (2019). Learning loss for active learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 93–102, Los Alamitos, CA, USA. IEEE Computer Society. 116, 121, 124

Yu, F., Zhang, Y., Song, S., Seff, A., and Xiao, J. (2015). Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *ArXiv*, abs/1506.03365. 116

Yu, L., Wang, S., Li, X., Fu, C.-W., and Heng, P.-A. (2019). Uncertainty-aware self-ensembling model for semi-supervised 3d left atrium segmentation. In *Medical image computing and computer assisted intervention–MICCAI 2019: 22nd international conference, Shenzhen, China, October 13–17, 2019, proceedings, part II 22*, pages 605–613. Springer. 135

Yu, Y., Wang, C., Fu, Q., Kou, R., Huang, F., Yang, B., Yang, T., and Gao, M. (2023). Techniques and challenges of image segmentation: A review. *Electronics*, 12(5):1199. 11, 12

Zaharia, M., Chen, A., Davidson, A., Ghodsi, A., Hong, S. A., Konwinski, A., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., et al. (2018). Accelerating the machine learning lifecycle with mlflow. *IEEE Data Eng. Bull.*, 41(4):39–45. 46

Zhan, X., Wang, Q., Huang, K.-H., Xiong, H., Dou, D., and Chan, A. B. (2022). A comparative survey of deep active learning. *ArXiv*, abs/2203.13450. 113, 116, 119

Zhang, C. and Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. *Journal of Industrial Information Integration*, 23:100224. 1

Zhang, Z., Wang, S., Li, Z., Gao, F., and Wang, H. (2023). A multi-dimensional covert transaction recognition scheme for blockchain. *Mathematics*, 11(4):1015. 34

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P. H., et al. (2021). Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890. 44

Zhou, B., Chen, L., and Wang, Z. (2019). Interactive deep editing framework for medical image segmentation. In Shen, D., Liu, T., Peters, T. M., Staib, L. H., Essert, C., Zhou, S., Yap, P.-T., and Khan, A., editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 329–337, Cham. Springer International Publishing. 89, 93

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929. 44

Zhou, C., Li, Q., Li, C., Yu, J., Liu, Y., Wang, G., Zhang, K., Ji, C., Yan, Q., He, L., et al. (2023a). A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *arXiv preprint arXiv:2302.09419*. 43

Zhou, S. K., Greenspan, H., Davatzikos, C., Duncan, J. S., Van Ginneken, B., Madabhushi, A., Prince, J. L., Rueckert, D., and Summers, R. M. (2021). A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. *Proceedings of the IEEE*, 109(5):820–838. 14, 16, 18

Zhou, S. K., Greenspan, H., and Shen, D. (2023b). *Deep learning for medical image analysis*. Academic Press. 19

Zhou, T., Li, L., Bredell, G., Li, J., and Konukoglu, E. (2022). Volumetric memory network for interactive medical image segmentation. *Medical image analysis*, 83. 89, 90, 99, 101, 106

Zhou, Z. (2023). Evaluation of chatgpt's capabilities in medical report generation. *Cureus*, 15(4). 20

Zhu, H., Meng, F., Cai, J., and Lu, S. (2016). Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *Journal of Visual Communication and Image Representation*, 34:12–27. 89

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2019). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109:43–76. 111

Zhuang, Y., Mathai, T. S., Mukherjee, P., and Summers, R. M. (2024). Segmentation of pelvic structures in t2 mri via mr-to-ct synthesis. *Computerized Medical Imaging and Graphics*, 112:102335. 28

Zoph, B., Ghiasi, G., Lin, T.-Y., Cui, Y., Liu, H., Cubuk, E. D., and Le, Q. V. (2020). Rethinking pre-training and self-training. *ArXiv*, abs/2006.06882. 89

Appendix A

Acronyms

AE Autoencoder. 35

AI Artificial Intelligence. 1, 2, 7, 137

AL Active learning. 36, 38, 113, 114, 116, 119, 140

ANN Artificial Neural Network. iii, 1

AR Augmented Reality. iii, 23, 25, 26, 138

BRS Backpropagating Refinement Scheme. 99, 140

CADe Computer-aided Detection. 19

CADx Computer-aided Diagnosis. 19

CAI Computer-Aided Intervention. 23

CAM Class Activation Mapping. 44

CHU Centre Hospitalier Universitaire. 29, 54

CIM Cumulative Interaction Memory. 88, 92, 93

CLIP Contrastive Language-Image Pre-training. 43

CNN Convolutional Neural Network. 11

CONSORT-AI Consolidated Standards of Reporting Trials. 49

CT Computed Tomography. 14, 16, 87

CV Coefficient of Variation. 67–70, 75, 77–82, 84

DCE Dynamic Contrast-enhanced Imaging. 27

DDG Dynamic Data Generation. 93–95, 105, 139–141

DICOM Digital Imaging and Communications in Medicine. 16

DL Deep Learning. iii, 2, 87–90, 98, 109, 138

DNN Deep Artificial Neural Network. 2, 4

DWI Diffusion-weighted Imaging. 27

ET Echo Time. 56

FC Fully Connected Neural Network. 13

FCN Fully Convolutional Neural Network. 13

FIGO International Federation of Gynaecology and Obstetrics. 41, 83

FMA Found Myoma Agreement. 68, 83

FPMRI Female Pelvis MRI. iii, iv, 27, 53, 93, 104, 108, 109, 111, 125, 129, 135, 137–139

FPMRId Female Pelvis MRI dataset. iv, 16, 53, 97, 98, 114, 121, 125, 126, 128, 129, 131–133, 135, 137, 139, 141

FSE Fast Spin Echo. 56

GAN Generative Adversarial Network. 35

GEOS Geodesic Image Segmentation. 12, 89

GPT-3 Generative Pre-trained Transformer 3. 28

GPT-4 Generative Pre-trained Transformer 4. 42

GPU Graphics Processing Unit. 34, 141

GRU Gated Recurrent Unit. 35

GUI Graphical User Interface. 10, 47, 102, 128

HDS Health Data Hosting Certification. 58

IPAT Integrated Parallel Acquisition Techniques. 56

LIME Local Interpretable Model-agnostic Explanations. 44

LLM Large Language Model. 28

LPH Loss Prediction Head. 120–122, 128, 130, 132, 133, 135

LPL Loss Prediction Loss. 121, 122

LSTM Long Short-Term Memory. 96

LUS Laparoscopic Ultrasound. 23

LV Left Ventricular. 134

LVIS Large Vocabulary Instance Segmentation. 39

MIS Minimally Invasive Surgery. 23

MITK Medical Imaging Interaction Toolkit. 47, 65

ML Machine Learning. iii, 1, 2, 4, 111–115, 117, 122

MLOps Machine Learning Operations. 45

MLP Multilayer Perceptron. 121

MONAI Medical Open Network for AI. 45

MRI Magnetic Resonance Imaging. 5, 14, 16, 87, 138

NAS Neural Architecture Search. 36

NLP Natural Language Processing. 36

NLST National Lung Screening Trial. 39, 41

PACS Picture Archiving and Communication System. 17

PCA Principal Component Analysis. 4

PET Positron Emission Tomography. 5, 14, 16

RJ Junior Radiologist. 68

RNN Recurrent Neural Network. 13

RR Radiology Resident. 68

RS Senior Radiologist. 68

RV Right Ventricular. 134

SAIM Single Active Interactive Model. iv, 31, 113, 114, 118, 119, 134, 135, 138–140, 142

SAM Segment Anything Model. 139

SDG Static Data Generation. 90, 103

SHAP Shapley Additive Explanations. 44

SIM Sequential Interaction Memory. 88, 92, 93, 121, 139–141

Slurm Simple Linux Utility for Resource Management. 46

SNN Shallow Artificial Neural Network. 2, 4

SOTA State of the Art. 6, 113, 134, 139, 140, 142

SSL Semi-supervised learning. 31, 36, 112, 114, 115, 135

ST Self-training. 31, 112, 114, 115, 135

STAPLE Simultaneous Truth and Performance Level Estimation. 30

STARD-AI Standards for Reporting Diagnostic Accuracy Studies. 49

STM Space-Time Memory Network. 141

T1WI T1-weighted imaging. 27

T2WI T2-weighted imaging. 27, 53

TCIA Cancer Imaging Archive. 29

timm Pytorch Image Models. 45

TR Repetition Time. 56

TRIPOD-AI Transparent Reporting of a Multivariable Prediction Model for Individual Prognosis or Diagnosis. 49

TSE Turbo Spin Echo. 56

UMAP Uniform Manifold Approximation and Projection. 64, 65

UMD Uterine Myoma MRI dataset. 39, 139

US Ultrasound. 14, 16

ViT Vision Transformer. 35

VR Virtual Reality. 25, 26

XAI Explainable AI. 42

xLSTM Extended Long Short-Term Memory. 141