Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité Informatique

Préparée au sein de INSA ROUEN NORMANDIE

Biomarker estimation from medical images: segmentation-based and segmentation-free approaches

Présentée et soutenue par

Jing ZHANG

Thèse soutenue publiquement le 06/04/2022 devant le jury composé de		
M. Désiré SIDIBÉ	PR de l'Université Evry Val Essonne	Rapporteur
M. Aymeric HISTACE	PR de l'ENSEA, Cergy	Rapporteur
Mme Mireille GARREAU	PR de l'Université de Rennes I	Examinatrice
Mme Caroline PETITJEAN	PR de l'Université de Rouen Normandie	Codirectrice de thèse
Mme Samia AINOUZ	PR de l'INSA Rouen Normandie	Directrice de thèse

Thèse dirigée par Samia AINOUZ et Caroline PETITJEAN, laboratoire LITIS

INSA ROUEN NORMANDIE

UNIVERSITÉ DE ROUEN NORMANDIE

MI/IΣ

litis

Acknowledgements

I would like to thank my dissertation reviewers for reading my PhD thesis and providing valuable comments and suggestions.

My 42-month PhD is coming to an end, and I could not have succeeded without the people who have helped me along the way.

First of all, I would like to sincerely thank my two lovely supervisors, professor Samia Ainouz and professor Caroline Petitjean who are from LITIS lab. They are great female scientists with the qualities of care, patience, and wisdom. They have given me very constructive guidance and encouragement in my work, which has made me have further insight and confidence in doing scientific research. In addition, their rigorous style in work and humor in life have deeply infected me. I am fortunate to be their student and hope I can learn from their good qualities.

During my first year as a PhD student, as I was new to France and not familiar with the language and environment, my supervisor Samia helped me find accommodation in advance with a French-speaking roommate and Caroline put me in an office with a Chinese colleague. My roommate Grace and my colleague Leo (Hongliu Cao) helped me through my first year of life and research respectively. I would also like to thank the leaders, teachers and other colleagues of the LITIS laboratory for their enthusiasm and commitment to my work, such as the secretaries Fabienne, Mathieu, Brigitte, technical staff Fabrice, colleagues Rosana, Wassim, Guillaume, Cyprien, etc (I wish I could remember everyone's name !).

As the saying goes, you can rely on your parents at home and your friends when you go out. I would like to thank my dear friends for their help and the happy times I had with them (Travelling, get-together, playing). They are Yang Ruiping, Wang Yanjun, Liu Xuelian, Lou Yuzhen, Jia Linlin, Wang Jundong, Xu Jie, Ren Chao, Wang Xuefei, Zhang Haodi, Zhang Ce, Yuan Wenbin, and due to space limitation, forgive me for not being able to list all the names of my friends.

As the Covid-19 pandemic began to spread across Europe in early 2020, coun-

tries began to close their cities, schools began to telecommute, and we were all suffering from varying degrees of physical and psychological discomfort. No one was more worried about me than my family, my parents and my brother and sister-in-law, who were thousands of kilometers away in China. From time to time, they were aware of the epidemic situation on my side and concerned about my health in body and mental, and it was a difficult time. My two adorable little nieces often tell me interesting stories about them, and I envy their childishness and carefree nature. During this time, the Chinese Embassy in France also cared about the students in France, they gave us “health kits" (containing masks, wet wipes, and some cold medicine) and the LITIS lab gave us a box of masks and hand sanitizer. In 2021, I got two doses of the Covid-19 vaccine for free, hoping that the epidemic would end sooner and everyone would return to their normal work and life as before. I’m so grateful and thankful for their support and care.

My PhD work was made possible thanks to the financial support from CSC (China Scholarship Council) and the convenient computational support from CRIANN (Le Centre Régional Informatique et d’Applications Numériques de Normandie) for research.

Finally, I would like to thank the Chinese and French organizations and people who made this UT-INSA project possible. Through this project, I had the opportunity to open a new world and experience the technology, humanities and cuisine (Les baguettes et les croissants sont mes préférés !) of France, as well as the surrounding European countries, which is a valuable spiritual treasure for me. There are many prejudices and misunderstandings in this world, and you can only gain your own insights if you experience and witness them yourself.

Résumé

La segmentation est l'une des tâches les plus importantes dans l'analyse des images médicales. Depuis quelques années, les réseaux de neurones convolutifs (CNN) en constituent l'état de l'art. Dans ce contexte, nous allons nous focaliser sur les problématiques suivantes. Premièrement, la fonction de perte (loss) est une composante importante qui dirige l'apprentissage des CNN et décide de la relation entre les étiquettes cibles et les prédictions. Les fonctions de loss standard en particulier, telle que la loss de Dice, ont montré leurs limites. Deuxièmement, la segmentation est souvent la première étape pour ensuite estimer les paramètres (également appelés biomarqueurs) de l'image. Ces biomarqueurs sont utilisés pour établir un diagnostic et un suivi des patients. Une estimation précise des biomarqueurs est donc capital. Cependant, des erreurs sont susceptibles de se produire lors de l'étape intermédiaire de segmentation. Récemment, les techniques d'apprentissage profond ont ouvert la voie à l'estimation directe des biomarqueurs à partir des images, sans segmentation ou extraction de caractéristique adhoc. La recherche sur ce sujet en est encore à ses débuts.

Pour répondre à ces questions, cette thèse propose les contributions suivantes, résumées en trois points : tout d'abord, nous proposons une nouvelle fonction de perte, basée sur le coefficient Kappa, qui a la capacité de prendre en compte tous les pixels de l'image, y compris le vrai négatif, contrairement à la perte standard de Dice. Nous illustrons sa valeur ajoutée sur un jeu de données public d'images de lésions cutanées. Deuxièmement, nous contribuons à la prédiction directe de biomarqueurs sans segmentation afin de fournir une solution d'analyse raisonnable et efficace pour les applications cliniques. Nous proposons plusieurs architectures de CNN de régression, qui apprennent directement à estimer les paramètres d'intérêt sans recourir à la segmentation. Un cas d'application est la prédiction de la circonférence de la tête du fœtus à partir d'images échographiques : nous comparons segmentation et régression avec un protocole expérimental judicieux. De ce fait, nous avons pu montrer des résultats prometteurs pour la régression, même si des améliorations restent possibles. Un autre cas est la prédiction des volumes de la structure cardiaque à partir d'images de résonance magnétique tridimensionnelles, dans lequel une méthode de prédiction multi-objectifs est réalisée. Troisièmement, nous étudions l'interprétabilité des modèles de régression, en étendant les techniques standard de cartes de saillance aux CNN de régression, qualitativement et quantitativement. Nous avons pu montrer que, sur la plupart des images, le CNN de régression apprend réellement à identifier la zone cible.

Mots-clés: Analyse d'image médicale, apprentissage profond, segmentation, fonction de perte, CNN régression, biomarqueurs, interprétabilité, circonférence de la tête du fœtus, volume des structures cardiaques.

Abstract

Segmentation is one of most prominent task in medical image processing and analysis. For a few years now, convolutional neural networks (CNN) have been the state-of-the-art in this domain. We will focus on CNN for medical image segmentation and analysis from the following standpoints. First, the loss function is an important component that drives the CNN training and decides on the relation between target labels and the predictions. As such, a lot of research is made on loss design, especially since the standard losses, such as the Dice loss, have shown their limitations. Second, segmentation is often the first step to subsequently estimate parameters (also called biomarkers) from the image. Medical experts use biomarkers to diagnose patients' health status and monitor treatment. Thus accurate biomarkers estimation is of paramount importance. However, errors are prone to occur in the intermediate segmentation step. Very recently, deep learning techniques have open the way to directly estimate biomarkers from images, without segmenting them. Research on this topic is still as its early stage.

To address the above issues, this thesis proposes the following contributions, summarized in three points : first, we propose a new loss function, that is based on the Kappa coefficient, that has the ability to take into account all the pixels in the image, including the true negative, contrary to the standard Dice loss. We illustrate its added value on a public set of skin lesion images. Second, we contribute to segmentation-free direct biomarker prediction, from a methodological perspective, so as to provide a reasonable and effective analysis solution for clinical applications. We propose and study several regression CNN architectures, that learn directly to estimate the parameters of interest without resorting to segmentation. One application case is the prediction of fetus head circumference (HC) from ultrasound images: we comprehensively compare segmentation-based method and regression (i.e. segmentation-free) method under a fair experimental protocol and are able to show promising results, even though room for improvement is left. Another case is prediction of cardiac structure volumes from 3-dimensional (3D) magnetic resonance images, in which a multi-objective prediction method is achieved. Third, we investigate the interpretability of the deep regression models, by extending standard saliency maps techniques to regression CNN. We explained the inner world of the regression CNN models both qualitatively and quantitatively and are able to show that indeed the regression CNN is learning to identify the target area.

Keywords: Medical imaging analysis, Deep learning, Segmentation, Loss function, Regression CNN, Biomarkers, Interpretability, Fetus head circumference, Cardiac structure volume

List of Abbreviations

Chapter 1

2/3 D	**2/3 D**imensional
CV	**C**omputer **V**ision
AI	**A**rtificial **I**ntelligence
DL	**D**eep **L**earning
MIC	**M**edical **I**maging **C**omputing
CT	**C**omputed **T**omography
MRI	**M**agnetic **R**esonance **I**maging
US	**U**ltra **S**ound
HC	**H**ead **C**ircumference
FL	**F**emur **L**ength
ISIC	**I**nternational **S**kin **I**maging **C**ollaboration
ACDC	**A**utomated **C**ardiac **D**iagnosis **C**hallenge
CVD	**C**ardio**V**ascular **D**iseases
LV	**L**eft **V**entricle
RV	**R**ight **V**entricle
MYO	**MYO**cardium
EF	**E**jection **F**raction
ED	**E**nd **D**iastole
ES	**E**nd **S**ystole
EDV	**ED V**olume
ESV	**ES V**olume
MLD	the **M**inimum **L**umen **D**iameter
RVD	**R**eference **V**essel **D**iameter
LL	**L**esion **L**ength
RAS	**R**enal **A**artery atherosclerosi**S**
CNN	**C**onvolutional **N**eural **N**etworks
RNN	**R**ecurrent **N**eural **N**etworks

NLP	**N**atural **L**anguage **P**rocessing
XAI	e**X**plainable **AI**

Chapter 2

NMR	**N**uclear **M**magnetic **R**esonance
fMRI	**f**unctional **MRI**
PET	**P**ositron **E**mission **T**omography
SPECT	**S**ingle **P**ositron **E**mission **C**omputerized **T**omography
FCN	**F**ully **C**onvolutional **N**etworks
GAN	**G**enerative **A**dversarial **N**etworks
LSTM	**L**ong **S**hort-**T**erm **M**emory
Seg	**Seg**mentation
GT	**G**round **T**ruth
TP	**T**rue **P**ositive
TN	**T**rue **N**egative
FP	**F**alse **P**ositive
FN	**F**alse **N**egative
HD	**H**ausdorff **D**istance
ASSD	**A**verage **S**ymmetric **S**urface **D**istance
BCE	**B**inary **C**ross **E**ntropy
WCE	**W**eighted **C**ross **E**ntropy
FL	**F**ocal **L**oss
DPCE	**D**istance map **P**enalized **C**ross **E**ntropy
SSE	**S**ensitivity-**S**pecificity **E**rror
IoU	**I**ntersection **o**ver **U**nion
GDL	**G**eneralized **D**ice **L**oss
TL	**T**versky **L**oss
FTL	**F**ocal **T**versky **L**oss
ASL	**A**symmetric **S**imilarity **L**oss
BL	**B**oundary **L**oss
HDL	**H**ausdorff **D**istance **L**oss
ELL	**E**xponential **L**ogarithmic **L**oss
ROI	**R**egion **O**f **I**nterest
HoG	**H**istogram **o**f **G**radients
SDL	**S**upervised **D**escriptor **L**earning
ANN	**A**rtifical **N**eural **N**etworks

SVM	**S**upport **V**ector **M**achine
MI	**M**yocardial **I**nfarction
LVSC	**L**eft **V**entricle **S**egmentation **C**hallenge
LVQUAN	**L**eft **V**entricle Full **QUAN**tification Challenge
GRU	**G**ated **R**ecurrent **U**nit
MAE	**M**ean **A**boslute **E**rror
PMAE	**P**ercentage **M**ean **A**boslute **E**rror
RMSE	**R**oot **M**ean **S**quare **E**rror
MBM	**M**uscle **B**ody **M**ass
FBM	**F**at **B**ody **M**ass
LBM	**L**ean **B**ody **M**ass
VAT	**V**isceral **A**dipose **T**issue
SAT	**S**ubcutaneous **A**dipose **T**issue
CIFAR	**C**anadian **I**nstitute **F**or **A**dvanced **R**esearch
CAM	**C**lass **A**ctivation **M**apping
LRP	**L**ayer **R**elevance **P**ropagation
LIME	**L**ocal **I**nterpretable **M**odel-**A**gnostic **E**xplanations
SHAP	**SH**apley **A**dditive ex**P**lanations
TCAV	**T**esting with **C**oncept **A**ctivation **V**ector
ECG	**e**lectro**c**ardio**g**ram
AOPC	**A**rea **o**ver **P**erturbation **C**urve

Chapter 3

DI	**D**ice **I**ndex
ReLU	**Re**ctified **L**inear **U**nit
SCD	**S**kin **C**ancer **D**etection

Chapter 4

pp	**p**ost **p**rocessing
EF	**E**llipse **F**itting
FPN	**F**eature **P**yramid **N**etworks
PSPNet	**P**yramid **S**cene **P**arsing **Net**work
VGG	**V**isual **G**eometry **G**roup
ResNet	**Res**idual **Net**work
EfficientNet	**efn**

MSE	**M**ean **S**quare **E**rror
HL	**H**uber **L**oss
Reg	**Reg**ression
GB	**G**iga**B**yte

Chapter 5

MINF	**M**yocardial **INF**arction
DCM	**D**ilated **C**ardio**m**yopathy
HCM	**H**ypertrophic **C**ardio**m**yopathy
ARV	**A**bnormal **R**ight **V**entricle
NOR	**Nor**mal cardiac

Appendix B

bs	**b**atch**s**ize
lr	**l**earning **r**ate
ViT	**Vi**sion **T**ransformer
Q	**Q**uery
K	**K**ey
V	**V**alue
MLP	**M**ulti **L**ayer **P**erceptron

Contents

List of Figures

List of Tables

List of Algorithms

Part I

Introduction and State of the Art

Chapter 1

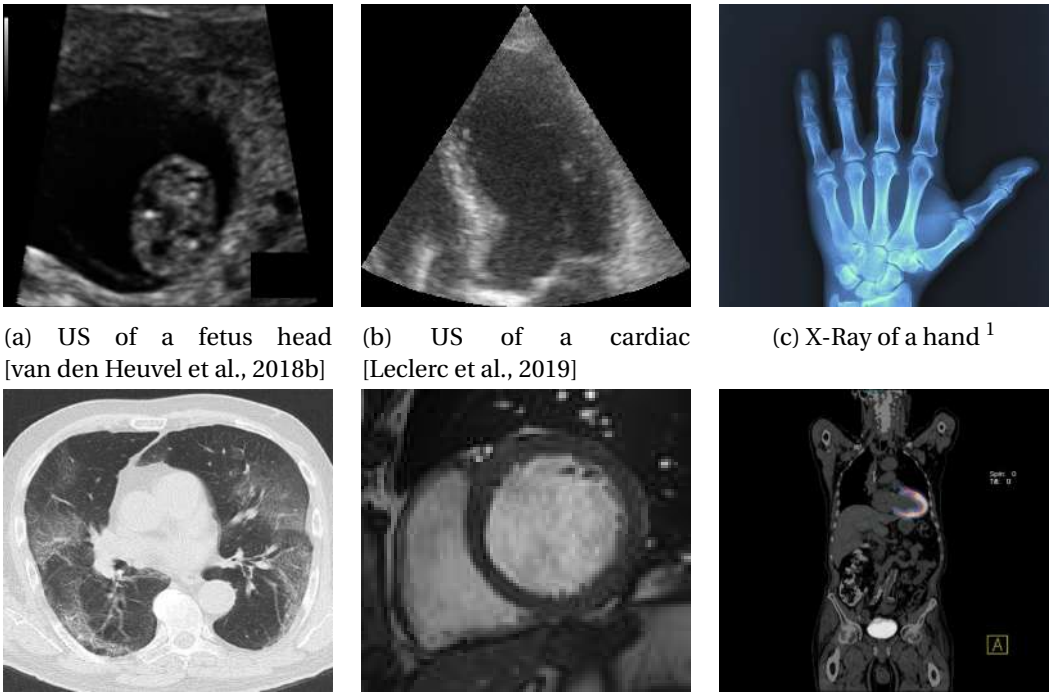
Introduction

Contents

1.1 Background

Computer science and technology has helped mankind increase productivity in every aspect since its inception. As a branch of it, computer vision (CV) tries to imitate the human eye to acquire, process, and analyze everything related to media, such as images, videos. Artificial intelligence (AI) emerged almost simultaneously with computer science. In the 21st century, techniques based on artificial intelligence have progressed considerably, and one specific typical approach that has been successful is deep learning (DL) techniques. Undoubtedly, computer vision based on deep learning techniques has contributed greatly in many fields [Chai et al., 2021]; for instance, medical image analysis or computing (MIC), which is the central theme to be highlighted in this thesis.

Medical image analysis is an interdisciplinary discipline that combines medical imaging and computer science. The most commonly used imaging modalities in clinical medicine include radiography (e.g. X-ray), computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound (US), among others. **X-Rays** (radiography) is first discovered in 1895. The principle is emitting electromagnetic waves to inside the human body, and the projection image is formed. Computed tomography (**CT scan**) developed in 1970s, uses multiple X-Ray machines in different angles to detect various parts (the soft tissues, blood vessels and bones etc.) of the body, then, reconstruct these images through computers to create cross-sectional images of the body. These images provide more detailed information than a normal X-ray image. Ultrasound (**US**) was first used for clinical purposes in 1956. Ultrasound is an imaging modality that uses high-frequency sound waves rather than radiation. The advantages of US images are real-time, fast, low cost, and not harmful to human. However, the quality of US images is bad, and noise is included sometimes. Magnetic Resonance Imaging (**MRI**) is a non-invasive imaging technology that produces three dimensional detailed anatomical images. The first MRI scan of the human body was performed in 1977. Based on the principles of nuclear magnetic resonance (NMR), MRI techniques use a strong magnetic field to force the protons inside a substance to align with that field. Based on the electromagnetic waves emitted by the decaying energy of the nucleus, the location and type of that nucleus can be known and an image is formed. One special type of MRI is functional MRI (fMRI), which is used to observe brain structure and determine which areas of the brain are "activated" when performing certain cognitive tasks. Thus, the brain



(a) US of a fetus head [van den Heuvel et al., 2018b]

(b) US of a cardiac [Leclerc et al., 2019]

(c) X-Ray of a hand [1]

(d) CT of a chest with COVID-19 [Kwee and Kwee, 2020]

(e) MRI of a cardiac [Bernard et al., 2018]

(f) PET of a body [2]

[1] The image is from https://www.imaginghealthcare.com/diagnostic-imaging/digital-x-ray/
[2] The image is from https://www.itnonline.com/article/what-pet-imaging

Figure 1.1 – Medical image types

organization can be understood through this way. Positron emission tomography (**PET**) is a nuclear imaging technology. The principal of PET is that the tracer is injected into a vein first, then PET systems detect and reconstruct the radiations from inside the body. Similar technology is Single Photon Emission Computerized Tomography (SPECT). And hybrid PET imaging systems (with CT or MRI) are practical in recent decades [Lee, 2010]. These five common modalities of medical imaging and their usages are summarized in Table 1.1 [1]. Besides the image types mentioned above, there are other types of images, such as skin lesion images, fundus images, histopathology images, etc. Some examples are given in Figure 1.1.

[1] The information is gathered from
`https://blog.radiology.virginia.edu/different-imaging-tests-explained/`

Table 1.1 – Five common modalities of medical imaging.

Modality	**Principle**	**Manner& Duration**	**Usage**
X-Ray	X-rays use ionizing radiation which are quick, painless tests that produce images of structures inside one's body, especially bones.	One will lie, sit, or stand while the x-ray machine takes images. One may be asked to move into several positions. 10-15 minutes.	• bone fractures • arthritis • osteoporosis • infections • breast cancer • swallowed items • digestive tract problems
CT Scan	CT scans use a series of x-rays to create cross sections of the inside of the body, including bones, blood vessels, and soft tissues.	One will lie on a table that slides into the scanner. The x-ray tube rotates around one to take images. 10-15 minutes.	• injuries from trauma • bone fractures • tumors and cancers • vascular disease • heart disease • infections • guide biopsies
MRI	MRIs use magnetic fields and radio waves to create detailed images of organs and tissues in the body.	One will lie on a table that slides into the MRI machine, which is deeper and narrower than a CT scanner. The MRI magnets create loud tapping or thumping noises. 45 minutes-1 hour.	• aneurysms • Multiple Sclerosis (MS) • stroke • spinal cord disorders • tumors • blood vessel issues • joint or tendon injuries
Ultrasound	Ultrasound uses high-frequency sound waves to produce images of organs and structures within the body.	A technician applies gel to one's skin, then presses a small probe against it, moving it to capture images of the inside of one's body. 30 minutes-1 hour.	• gallbladder disease • breast lumps • genital/prostate issues • joint inflammation • blood flow problems • monitoring pregnancy • used to guide biopsies
PET Scan	PET scans use radioactive drugs (called tracers) and a scanning machine to show how one's tissues and organs are functioning.	One will swallow or have radiotracer injected. One then enter a PET scanner (which looks like a CT scanner) which reads the radiation gives off by the radiotracer. 1.5-2 hours.	• cancer • heart disease • coronary artery disease • Alzheimer's Disease • seizures • epilepsy • Parkinson's Disease

Imaging data accounts for approximately 90% of all medical data and is therefore one of the most important sources of evidence for clinical analysis and medical intervention [Zhou et al., 2021]. The goal of medical image analysis is developing computational and mathematical methods to solve problems related to medical images and use them in biomedical research and clinical care [Wikipedia, 2021]. This field involves several broad tasks: image segmentation, image registration, image classification, etc.

Nowadays, deep learning techniques have been successfully applied in different medical imaging analysis tasks, such as image classification, image segmentation, image registration, image reconstruction, object detection, etc. Medical image analysis based on AI can extract useful information from images, which can help doctors or experts to diagnose or make decisions about patients. If medical image analysis is aided by AI, it can greatly reduce the amount of effort doctors spend on a patient, especially in less developed areas where medical resources are not sufficient [Vuong et al., 2019]. Therefore, it is a very meaningful thing in terms of research and clinical applications. At the same time, one should also be wary of whether current AI technology (represented by DL) is safe and reliable in facing sensitive subjects with the AI techniques prospering, and why it makes this or that decision. In other words, these deep learning models should be developed with great reliability and transparency in sensitive areas such as medicine or autonomous driving. Consequently, there is a branch of AI called explainable AI (XAI) [Samek et al., 2021], which is aiming to make the AI reliable and trustworthy.

1.2 Motivation

In this thesis, we focus on medical image segmentation. Image segmentation is the process of partitioning the image into meaningful regions. In medical imaging, segmentation is often the first step required to estimate parameters (also called biomarkers) from the image, such as the volume of the segmented region, and is one of the major task in medical image analysis, useful for computer-aided patient diagnostic, pronostic and follow-up. More specifically, we address 3 specific issues. The first issue is the class imbalance problem in supervised learning that occurs in medical image segmentation; the second issue is biomarker estimation from medical imaging based on deep learning; the third issue is the explainability of some deep learning model that is applied in medical imaging analysis.

First of all, the deep learning techniques are widely used in various fields. It is a data-driven, automated predictive machine. The architectures of DL are evolving rapidly with supervised learning, unsupervised/semi-supervised learning, transfer learning, federated learning, etc. Specifically, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), attention mechanism and other models are widely used according to different needs. Recently, the Transformer networks [Vaswani et al., 2017] are quite popular in Natural Language Processing (NLP) and computer vision. No matter in which method above, the loss function is an important and integral part of neural networks. For instance, in supervised learning, the loss function is used in the neural networks to update the weight parameter of each neuron in back propagation stage, thus closing the gap between predicted and target values. The better the loss function is, the successful the performance of the model is usually, making other variables more consistent. In general, there are several types of loss functions in image segmentation, including Cross Entropy loss series and Dice losses which are derived from evaluation metrics [Ma et al., 2021]. In medical image segmentation, one prominent issue is the class imbalance problem, which refers to the ratio of foreground (segmentation target) and background in an image is severely unbalanced. For example, when segmenting a tumor from organ image or a lesion from skin image, in which the tumor or lesion is far smaller than the background (See Figure 1.2). So in this case, even though the segmentation results is not well matched the ground truth, the accuracy can still be high, because the model incorrectly takes into account the correct prediction of the background to count as the accuracy. Therefore, to this end, this thesis tries to find an optimization scheme i.e. loss function that can avoid the class imbalance problem and thus can really improve the image segmentation accuracy.

Secondly, the biomarker is a vital concept in clinical examination and diagnosis. Broadly speaking, the definition of a biomarker [Califf, 2018] is deceptively simple: "A defined characteristic that is measured as an indicator of normal biological processes, pathogenic processes or responses to an exposure or intervention." Specifically, there are two categories of biomarkers: imaging biomarkers and molecular biomarkers. Obtaining a biomarker from a medical image is relatively straightforward and easy, whereas obtaining the biomarker at the molecular level requires rigorous biochemistry-based experiments. In general, some known biomarkers are mainly achieved by two steps, which are segmentation step and geometry computation based on segmentation results. That is to say, medical image segmenta-

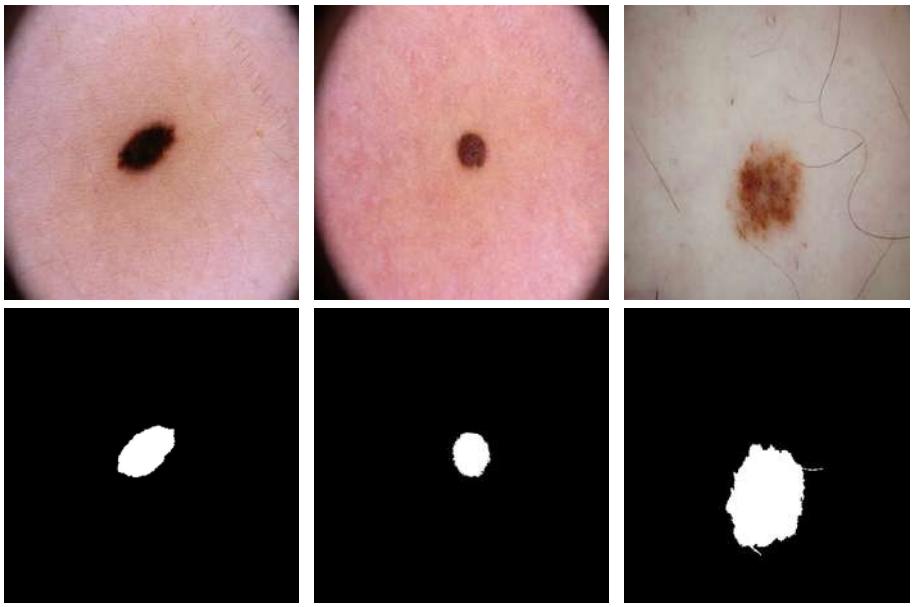


Figure 1.2 – Skin lesion images. The top images are skin photos, the bottom images are ground truth. The images are from Public ISIC 2018 dataset [Codella et al., 2018].

tion is only an intermediate step, and its further goal is to compute some kind of biomarker [Califf, 2018] or to serve image classification for determining which disease is present. Moreover, the problem associated with segmentation methods is that they are prone to errors and take an extra post processing steps and biomarker computation. This thesis is thus dedicated to exploring the feasibility to implement a direct prediction biomarker method so that it can bypass the segmentation-based approaches, which the regression-based methods just fit this scenario. Regression CNNs were first implemented for head pose estimation and facial landmark detection [Riegler et al., 2013, Ahn et al., 2014]. Afterwards, this idea has been applied in medical imaging analysis in order to solve different kinds of medical data and improve the performance as well as possible; for example, for left ventricular volumes prediction[Luo et al., 2016, Degrave et al., 2016, Ge et al., 2019c], mitosis counting for breast cancer diagnosis [Chen et al., 2016], aortic diameters estimation [Fernández, 2021], carotid artery indices estimation [Zhao et al., 2021].

The specific application that we will target in this thesis is examination of fetus growth and development during pregnancy. Head circumference (HC) is one of the key indexes to check a fetus growing state in clinical diagnose. Figure 1.3 is an example of fetus head in the form of ultrasound images. The fetus head is approximated as an ellipse annotated by experienced sonographers. Generally, a fetus growth is

divided into three trimesters [van den Heuvel et al., 2018a] according to the length of head circumference. With the aid of deep learning techniques, the segmentation of head circumference becomes efficient and accurate, but post-processing of the segmentation results, i.e., ellipse fitting and perimeter calculation, is still required. Therefore, this thesis is aiming to use a segmentation-free method to directly predict HC.

Figure 1.3 – US images of fetus head, the red ellipses are ground truth annotated by sonographers, below the images are the values of head circumferences in millimeter (mm) and pixels. The images are from Public HC18 dataset [van den Heuvel et al., 2018b].

Another research case is the advance screening and diagnosis of cardiovascular diseases with the help of deep learning techniques. Cardiovascular diseases (CVD) are common among all the diseases, which is the leading cause of death globally, killing an estimated 17.9 million people each year (the information is from World Health Organization.). Therefore, its importance and urgency has attracted countless studies from various aspects. For example, Figure 1.4 shows a group of MR cardiac images in short-axis view and their ground truth, which including the left ventricle (LV) in white color, right ventricle (RV) in gray color , myocardium (MYO) in light gray color, as well as the two states of cardiac: end systole (ES) and end diastole (ED), these indices are important for cardiac diseases diagnose. In clinical medicine, a vital criteria called ejection fraction (EF) which means the rate of the blood pump with heart beat of left ventricle or right ventricle defined in Equation 1.1:

$$\mathrm{EF}(\%) = \frac{\mathrm{EDV} - \mathrm{ESV}}{\mathrm{EDV}} * 100 \tag{1.1}$$

in which EDV means the volume of LV or RV in ED phase, ESV means the volume of LV or RV in ES phase. EF can reflect if the heart of a person is normal or not. Healthy people have ejection fractions between 50% and 65% [Kumar et al., 2014]. If one's EF is lower than normal index, which means that ejection volume is low, then

it could be heart failure, which may be caused by abnormal contraction or diastole of the heart. Generally, the volume of LV or RV is obtained also through two steps or more, which are performing segmentation, and then geometric computation. Since the cardiac data are composed of multiple slices scanned by the MR machine, the volume is usually calculated by accumulating the area of the cardiac structure slice by slice. The goal of this thesis is to explore a scheme of directly prediction the cardiac structure volumes without segmentation intervention.

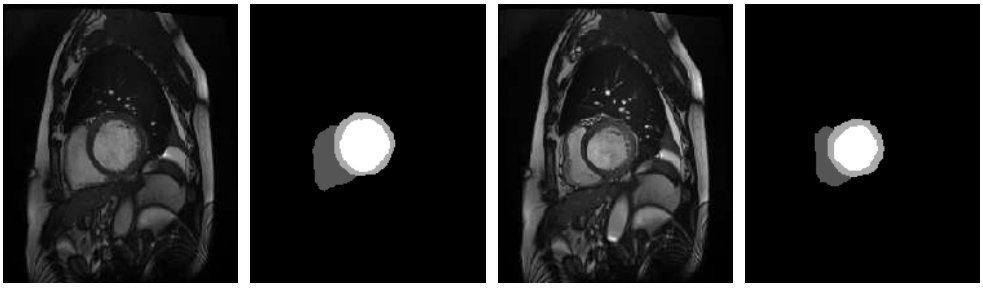


Figure 1.4 – Cardiac structure MR images. The first two images are one slice of a patient in ED stage and its ground truth. the last two images are one slice of the same patient in ES stage and its ground truth. The images are from Public ACDC dataset [Bernard et al., 2018].

Additionally to the cases described above, there are many other medical studies. For instance, the researches about coronary artery from X-ray image. It's important to know the specific indices/biomarkers of coronary artery for diagnosing the disease of patients, such as diameters (the minimum lumen diameter, MLD; reference vessel diameter, RVD) and lengths (lesion length, LL) of these vessels [Zhang et al., 2019]. Changes in kidney volume may reflect whether it is functioning properly. Based on this criterion, clinical medicine has defined a kidney disease called renal artery atherosclerosis (RAS) [Hussain et al., 2016]. Therefore, it is necessary to estimate the volume of the kidney from the scanned images with the help of segmentation or segmentation-free methods. The problem of adolescent scoliosis has also attracted a lot of attention in recent years. The scoliosis diagnosis is generally based on the idea of Cobb angle, which is defined as the largest angle at a particular region of the vertebral column [Sun et al., 2017].

Hence estimation of biological indicators has a great demand in clinical medicine. On the top of that, direct prediction-based methods are beginning to emerge with an accuracy that remains to be fairly compared to the accuracy of segmentation-based methods. Therefore, it's necessary to evaluate and compare these two kinds of methods from methodological and practical perspectives.

Finally, deep learning models have long been known for their groundbreaking performance. However, DL models are used like a black box; little is known about the decision process inside the DL model. In other words, the DL models should become more explainable or interpretable when making decisions on specific tasks [Rudin, 2019]. If this technology is to be implemented into practical applications, such as smart healthcare, autonomous driving and other cutting-edge areas, then it must be understandable and trustworthy, otherwise it could lead to fatal accidents. In particular, in the segmentation-free approach, we cannot visualize the prediction results like in the segmentation-based approach. This requires that the segmentation-free model is evidence-based when making decisions. Therefore, this study attempts to make an interpretation of the deep learning model according to the specific medical imaging problem.

1.3 Contributions of the research

A new metric-based loss function

We proposed a new metric-based loss function, called Kappa loss, which considers all the pixels including background information that Dice loss ignores, the proposed loss function is proved to be reasonable and superior to Dice loss both in theoretical and experimental (on several skin lesion datasets) aspects.

Direct biomarker prediction using regression CNNs

We proposed a direct fetus head circumference prediction method (regression CNNs) from ultrasound images that bypasses the segmentation based approaches on the public HC18 dataset [van den Heuvel et al., 2018b]. We compared the segmentation-free methods with the segmentation-based methods in a fair experimental environment.

We utilized regression CNN model to directly predict the three volumes of cardiac structures simultaneously from 3D magnetic resonance images on the public ACDC dataset [Bernard et al., 2018], multiple-channel based transfer learning was achieved on 3D medical images. To address the problem of insufficient data, data augmentation based on grid search is applied. Moreover, we performed cardiac data preprocessing including data cropping and slice number unifying and statistically analyzed and discussed the prediction results.

The explainability of regression CNN

We explained the black box of regression CNNs by several explaining methods in the forms of saliency maps and quantitative results. Besides, we achieved a customized evaluation metrics based on perturbation to quantitatively criticise different explaining methods on regression CNNs. The contributions of this thesis is concluded in Figure 1.5.

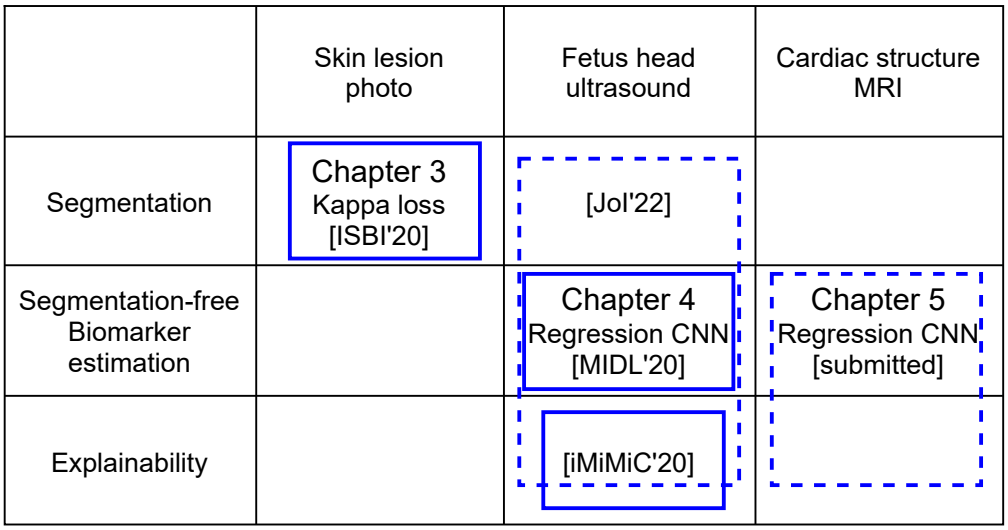


Figure 1.5 – Contributions of this thesis. Three techniques with respect to three kinds of medical image data (Application cases).

1.4 Structure of the thesis

The structure of the thesis is organized as follows:

Chapter 2 introduces the state of the art of the medical image segmentation methods, direct biomarker estimation methods, and explainable AI.

Chapter 3 describes the proposed Kappa loss function.

Chapter 4 presents the work of fetus head circumference prediction.

Chapter 5 focuses on the multi-structure of cardiac volume prediction.

Chapter 6 concludes the thesis and provides perspectives on future work.

The organization of this thesis is shown in Figure 1.6.

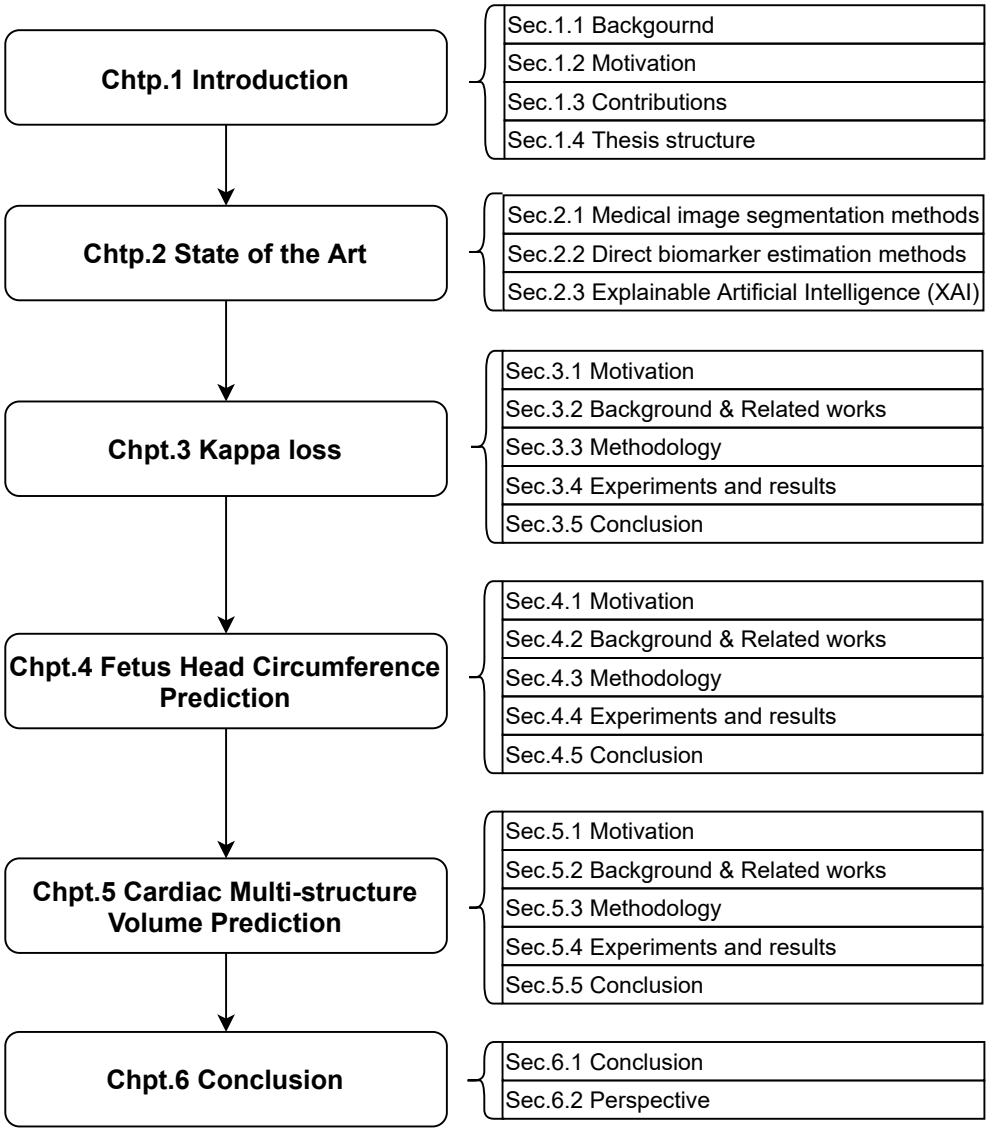


Figure 1.6 – Thesis structure.

List of Publications

During this Ph.D., the following research works have been published:

Peer-reviewed journal papers

Zhang, Jing, Caroline Petitjean, and Samia Ainouz. 2022. "Segmentation-Based vs. Regression-Based Biomarker Estimation: A Case Study of Fetus Head Circumference Assessment from Ultrasound Images" Journal of Imaging 8, no. 2: 23. https://doi.org/10.3390/jimaging8020023

Peer-reviewed international conference papers

Jing Zhang, Caroline Petitjean, and Samia Ainouz. "Kappa loss for skin lesion segmentation in fully convolutional network." 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI). IEEE, (2020): 2001-2004.

Jing Zhang, Caroline Petitjean, Pierre Lopez, and Samia Ainouz. "Direct estimation of fetal head circumference from ultrasound images based on regression CNN." Medical Imaging with Deep Learning (2020): 914-922.

Jing Zhang, Caroline Petitjean, Florian Yger, and Samia Ainouz. "Explainability for regression CNN in fetal head circumference estimation from ultrasound images." Interpretable and Annotation-Efficient Learning for Medical Image Computing (2020): 73-82.

Chapter 2

State of the Art

Contents

This chapter presents three states of the art in three domains of interest: medical image segmentation, direct biomarker estimation from medical images, and explainable AI for computer vision models in deep learning. For each of this field, we also provide the evaluation metrics and tools used to assess the methods.

2.1 Medical image segmentation methods

2.1.1 Traditional medical image segmentation methods

The definition of segmentation is subdividing an image into its constituent parts that are homogeneous in certain feature [Ramesh et al., 2021]. Traditional segmentation methods can be divided into the following categories:

Threshold segmentation As the name implies, is an algorithm that divides the image into two parts (background and foreground) based on a pixel threshold given in advance. Otsu's method [Otsu, 1979] is the representation of this idea.

Region-based methods Three methodologies are included in this scope. One is region growing algorithm [Adams and Bischof, 1994]. A seed point and similarity criteria decide the segmentation result. The other one is region split and merge algorithm [Chen and Pavlidis, 1979]. The image is divided into 4 pieces, and if one of these pieces meets the splitting conditions, then this piece is split into 4 pieces again, and so on. When the number of splits reaches a certain level, the adjacent blocks are merged if they meet certain conditions. The third one is watershed approach [Serge and Lantuéj, 1979]. The idea is that low-intensity pixels are regarded as valleys of the surface, high-intensity pixels are peaks. When the level rises to a certain height, water overflows the current valley. This can be achieved by building dams on the watershed, thus avoiding the pooling of water from both valleys, so that the image is divided into 2 sets of pixels, one for the valley flooded by water and one for the watershed line pixels. Eventually the lines formed by these dams then partition the whole image and achieve segmentation of the image.

Clustering methods Clustering is the partitioning of a data set into different classes or clusters according to a specific criterion (e.g. distance), so that the similarity of data objects within the same cluster is as large as possible, while the difference of data objects not in the same cluster is also as large as possible. The classical clustering algorithm is the K-Means algorithm [Hartigan and Wong, 1979].

Edge detection It is a fundamental problem in image processing and computer vi-

sion. The purpose of edge detection is to identify points in a image that have significant changes in intensity. The edge is formed by separating two areas according to distinct intensity. Mathematically, edge detection is roughly the calculation of the derivative of brightness change. Once we have calculated the derivatives, the next step is to give a threshold to determine where the edges are located. Commonly used algorithm is Canny algorithm [Canny, 1986].

Graph theory based segmentation The idea of this type of methods is to transform the pixel points of an image and their neighbors into vertices and edges and weights on edges in graph theory. Graph cuts and Grab cuts [Rother et al., 2004] are two examples. They utilize min cut algorithm to cut the edges connected between foreground and background.

Energy optimization algorithm The basic idea is to use a continuous curve to express the target edge and define a generalized energy function so that the independent variable includes the edge curve, so the segmentation process is transformed into the process of solving the minimum value of the generalized energy function, which can be generally achieved by solving the Euler equation corresponding to the function (Euler Lagrange) equation, the position of the curve where the energy is minimized is where the target profile is located. According to the different forms of curve expression in the model, the active contour models can be divided into two categories: parametric active contour model (Snake model [Kass et al., 1988]) and geometric active contour model (Level set method [Malladi et al., 1995]).

2.1.2 Deep learning based image segmentation

In recent decade, the deep learning techniques have been a great success due to the excellent performance than the traditional approaches in computer vision, natural language processing, etc [O'Mahony et al., 2019]. Deep learning essentially consists of data and models that depend on each other, and scholars have designed variety of deep neural network models with different learning abilities based on their own characteristics of data, especially medical data. This section reviews the deep learning based image segmentation according to the following figure (Figure 2.1).

Model architecture in image segmentation

Fully Convolutional Neural Networks, FCN [Long et al., 2015] which was the first proposed in image segmentation. The model fuses the shallow layers and deep

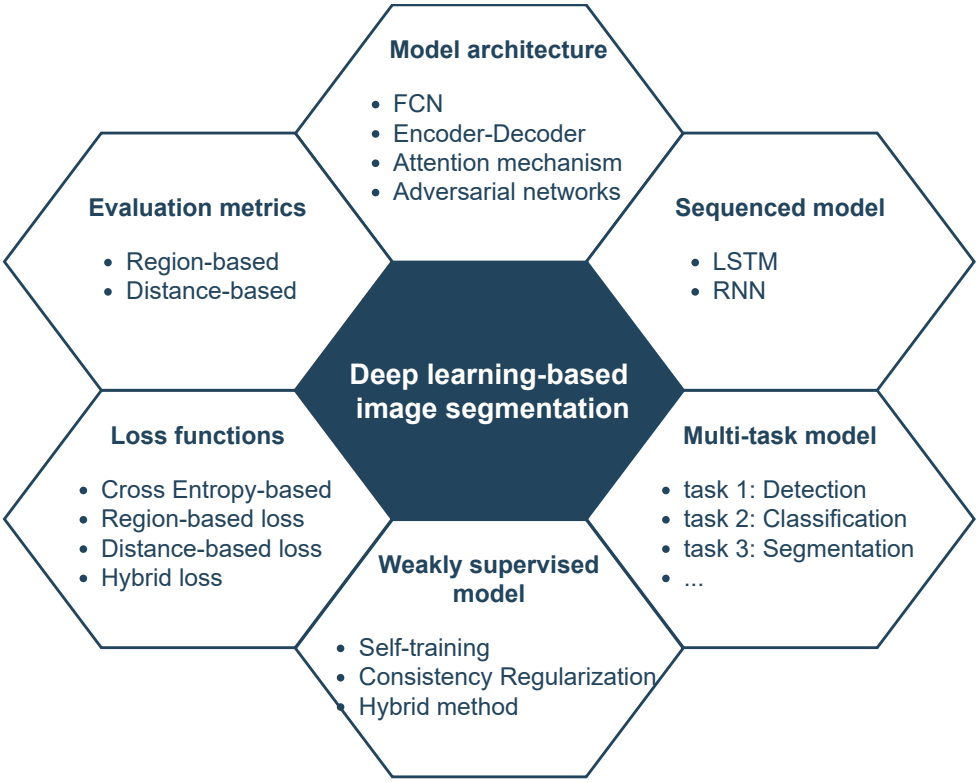


Figure 2.1 – Review of deep learning based image segmentation (Please start reading from 12 o'clock and in clockwise direction).

layers to preserve the the contextual spatial information. The U-Net is a FCN variant that has with symmetric encoding-decoding path, and skip connections [Ronneberger et al., 2015]. It is the most popular image segmentation model (Figure 2.2). In this supervised mode, training images and corresponding labels/ground truth are fed into the model. Each layer has number of convolutional filters/kernels, following with activation function as well as pooling operation in order to form feature maps. The weights of the neurons are obtained by the back-propagation of the error between the predicted value and the groud truth value, monitored by the learning rate. The loss function judges how well or how close the predicted value is from ground truth value, for example Cross Entropy loss is used in U-Net. Besides, this kind of model is actually data-driven, that is to say, a model will have a robustness and generalizability when training with a great deal of data. Thus, data augmentation is usually needed to increase the quantity of the data. The decoder of segmentation models is used to restore the segmentation map for final output, it can be deconvolution or upsampling layer, the difference is with trainable parameters or not. As for the output segmentation map, it is actually the pixel-wise

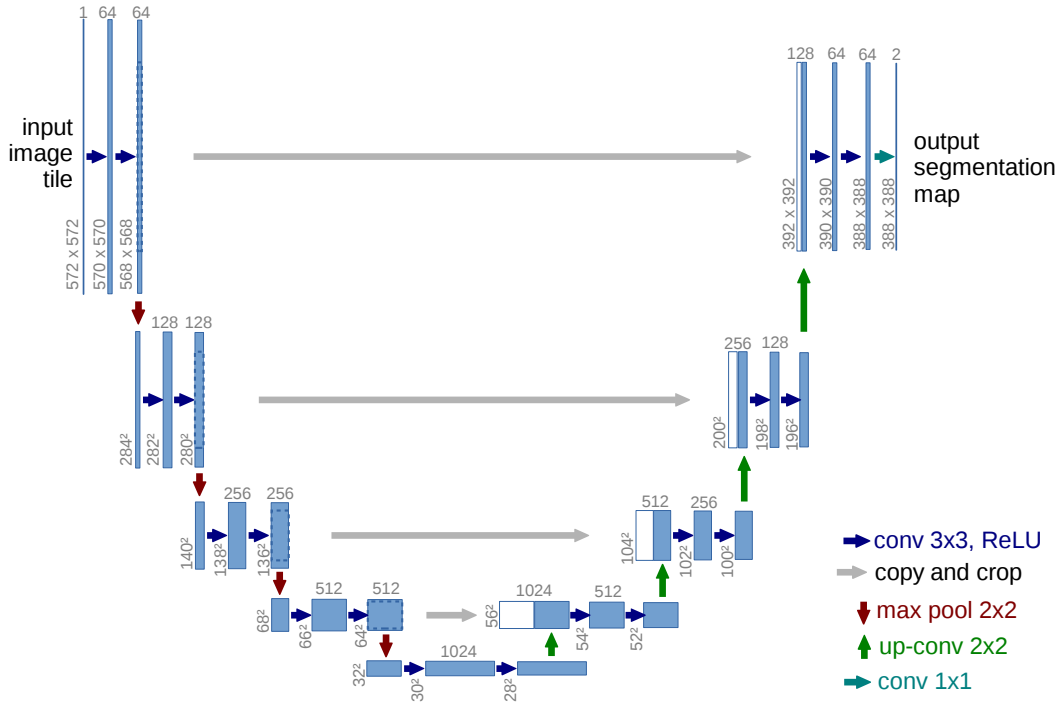


Figure 2.2 – Original U-Net architecture. The number of channels is denoted above the box, different colors mean different operation.

probability of each class generated by activation function Sigmoid ($\sigma = \frac{1}{1+e^{-x}}$) or Softmax ($s = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$), where x is element of input vector, K is the number of classes in multi-class classifier.

Then, 3D U-Net [Çiçek et al., 2016] and V-Net [Milletari et al., 2016] were proposed to deal with 3D medical data. Both these two models have skip connections among encoder and decoder. Similar encoder-decoder models are SegNet [Badrinarayanan et al., 2017], LinkNet [Chaurasia and Culurciello, 2017], Deeplab [Chen et al., 2017], PSPNet [Zhao et al., 2017], FPN [Lin et al., 2017a], U-Net++ [Zhou et al., 2018], Double U-Net [Jha et al., 2020], etc.

Attention-based models In psychology, attention is the cognitive process of selectively focusing on one or several things at the expense of others. The attention mechanism was first used in natural language processing. Then the Transformer [Vaswani et al., 2017] with only attention mechanism makes further progress. In computer vision including image segmentation, attention mechanisms are also starting to come to the fore, such as Squeeze-and-excitation networks [Hu et al., 2018], Attention U-Net [Oktay et al., 2018], Vision Transformer [Dosovitskiy et al., 2020], Swin Transformer [Liu et al., 2021a], and the very recent

ConvNeXT [Liu et al., 2022].

Models for image sequence

In medical image segmentation, some data are based on time series, for instance, the state of the heart/cardiac is different in each frame, and if we want to know the two states of the heart in systole and diastole, we have to find out these two states from different frames. To this end, the Recurrent Neural Networks (RNN) [Rumelhart et al., 1986] and long short-term memory (LSTM) neural networks [Hochreiter and Schmidhuber, 1997] can be applied for extraction of spatial and temporal information from specific medical data and tasks.

Multi-task models

Multi-task learning [Caruana, 1997] is a machine learning method based on shared representation, where multiple related tasks are put together to learn. And the purpose of the shared representation among different tasks is to improve generalization. The concept of multi-task learning exists because previous models have been single-task learning. In medical image analysis, tasks such as object (organs) detection, segmentation (lesion), regression, classification (disease) have been achieved by multi-task models [Zhang et al., 2012, He et al., 2019, Si and Roberts, 2019, Lian et al., 2021, Jia et al., 2021].

Weakly supervised models

Due to the annotation of medical images is time-consuming and laborious in reality. Therefore, the weakly/semi-supervised or unsupervised learning model [Chapelle et al., 2009] is an expedient way to compensate for the situation where there is only a small amount of labeled data and a large amount of unlabeled data, hoping to achieve the same or similar learning results as supervised learning with a fully labeled dataset. The overall idea of semi-supervised deep learning covers three types of architectures [Chaudhary, 2020]:

- Self training.
  - Pseudo-label [Lee et al., 2013]
  - Noisy Student [Xie et al., 2020c]
- Consistency Regularization.

  - π-model [Laine and Aila, 2017]
  - Temporal Ensembling [Laine and Aila, 2017]
  - Mean Teacher [Tarvainen and Valpola, 2017]
  - Virtual Adversarial Training [Miyato et al., 2018]
  - Unsupervised Data Augmentation [Xie et al., 2020b]
- Hybrid Method (Combining self training and consistency regularization).
  - MixMatch [Berthelot et al., 2019]
  - FixMatch [Sohn et al., 2020]

2.1.3 Loss functions

The loss function (also called cost or objective function) is one of key components in deep learning models that drives the optimization of the neural networks. Because it dictates how the error between the predicted value and the ground truth is computed and backpropagated throughout the networks. In this section, we will introduce 4 types of loss functions, namely Cross Entropy based loss, region based loss, distance based loss as well as hybrid loss.

Cross Entropy based loss

Cross Entropy loss It describes the distance between two distributions. The smaller the Cross Entropy, the closer the two are. Taking binary pixel/image classification as an example, one distribution is the class prediction probability, which is the output (p_i) of Sigmoid ($p_i = \frac{1}{1+e^{-x_i}}$) or Softmax ($p_i = \frac{e^{x_i}}{\sum_{j=1}^{C} e^{x_j}}$) in multi-class (C), x is weight value of each pixel/neuron (i). The other one is the corresponding class ground truth (g_i). Thus, the average of Binary Cross Entropy loss (BCE) over N pixels in an predicted image is composed by foreground and background two parts:

$$\mathrm{BCE} = -\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} [g_i \cdot log(p_i) + (1 - g_i) \cdot log(1 - p_i)] \tag{2.1}$$

If the background is much larger than the foreground, the loss is still small even though the segmentation result is inaccurate. This is the so-called class imbalance problem. To solve this class problem, then different weights need to be set on different terms.

Weighted Cross Entropy The wighted cross-entropy (WCE) has been used in

[Ronneberger et al., 2015]. The two-class form of WCE can be expressed as

$$\mathrm{WCE} = -\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} \omega g_i \log(p_i) + (1-g_i)\log(1-p_i), \tag{2.2}$$

where $\omega = (\mathrm{N} - \sum_{i=1}^{\mathrm{N}} p_i)/\sum_{i=1}^{\mathrm{N}} p_i$, which is the weight of foreground class. ω is inversely proportional to the class frequency in order to penalize the major class (in this case is the background).

Focal loss Focal loss (FL) [Lin et al., 2017b] is the variant of Cross Entropy loss. It solved the extreme object-background class imbalance problem by adding two coefficients α and γ to balance the weight of one-class examples, and adjust the rate to increase the importance of correcting mis-classified examples. In the original paper, the best performance was when the γ value was set to an empirical value of 2, $\alpha = 0.75$.

$$\mathrm{FL} = -\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} [\alpha \cdot g_i^{\gamma} \cdot \log(p_i) + (1-\alpha)\cdot(1-g_i)^{\gamma}\cdot\log(1-p_i)] \tag{2.3}$$

Distance map penalized cross entropy loss (DPCE), it [Caliva et al., 2019] is also a variant of Cross Entropy loss. In Equation 2.4, D is the distance penalty term of foreground class, specifically, D is euclidean distance matrix [1] of the ground truth. And $\odot$ is the Hadamard product [2]. In this way, pixels on the boundary can be given greater weights.

$$\mathrm{DPCE} = -\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} (1+\mathrm{D}ist(g_i)) \odot g_i \log(p_i) + (1-g_i)\log(1-p_i) \tag{2.4}$$

Region based loss

Sensitivity-Specificity error (SSE) This loss function [Brosch et al., 2015] combines mean squared difference between lesion region (sensitivity) and non-lesion region (specificity), regularized by a parameter r to control the ratio between this two parts. The benefit of mean squared errors is generating smooth gradients, so that making robust optimization results.

$$\mathrm{SSE} = r\frac{\sum_{i=1}^{\mathrm{N}}(p_i-g_i)^2 p_i}{\sum_{i=1}^{\mathrm{N}} p_i} + (1-r)\frac{\sum_{i=1}^{\mathrm{N}}(p_i-g_i)^2(1-p_i)}{\sum_{i=1}^{\mathrm{N}}(1-p_i)} \tag{2.5}$$

[1] Please refer to scipy.ndimage.morphology.distance_transform_edt.

[2] The Hadamard product operates on identically shaped matrices and produces a third matrix of the same dimensions.

Dice loss It originates from the Dice coefficient [Dice, 1945] which calculates the overlap between ground truth and the segmented image. If the Dice score is 1, which indicates that the predicted image matches perfectly with ground truth data. Here, in order to make loss converge, let the Dice be negative and plus 1. It was first used in V-Net [Milletari et al., 2016], now it has been widely used in medical image segmentation tasks.

$$\mathrm{DICE} = 1 - \frac{2\sum_{i=1}^{\mathrm{N}} p_i g_i}{\sum_{i=1}^{\mathrm{N}} (p_i + g_i)} \tag{2.6}$$

IoU loss Intersection over Union (IoU) loss [Rahman and Wang, 2016] is similar to the Dice loss, also called Jaccard loss, which is defined as:

$$\mathrm{IoUL}oss = 1 - \frac{\sum_{i=1}^{\mathrm{N}} p_i g_i}{\sum_{i=1}^{\mathrm{N}} (p_i + g_i - p_i g_i)} \tag{2.7}$$

Generalized Dice loss The authors [Sudre et al., 2017] add weights on the Dice loss for multi-class segmentation problem. The weight ($w = 1/(\sum_{i=1}^{\mathrm{N}} g_i)^2$) is inversely proportional to the ratio of that class.

$$\mathrm{GDL} = 1 - \frac{2\sum_{c}^{\mathrm{C}} w_c \sum_{i=1}^{\mathrm{N}} p_i g_i}{\sum_{c}^{\mathrm{C}} w_c \sum_{i=1}^{\mathrm{N}} (g_i + p_i)} \tag{2.8}$$

Lovász loss The idea of Lovász loss [Berman et al., 2018] is actually the IoU loss or Jaccard loss, but they use smooth extensions by Lovász extension in Convex optimization to deal with discrete problem of IoU loss. Specifically, first, they compute the the misclassified pixels (m):

$$m_i = \begin{cases} 1 - p_i, & \text{if} \quad g_i = 1 \\ p_i, & \text{otherwise} \end{cases} \tag{2.9}$$

Second, because the Jaccard loss Δ_{J} is submodular, then the Lovász extension can be used to compute the loss.

$$\Delta_{\mathrm{J}} = 1 - \mathrm{IoU} = \frac{\sum_{i=1}^{\mathrm{N}} m_i}{\sum_{i=1}^{\mathrm{N}} g_i \cup \sum_{i=1}^{\mathrm{N}} m_i} \tag{2.10}$$

$$\mathrm{L}ovász\ \ loss : \overline{\Delta_{\mathrm{J}}} = \sum_{i=1}^{\mathrm{N}} m_i del_i(\mathrm{S}orted(m_i)) \tag{2.11}$$

with $del_i(\cdot) = \Delta(\cdot) - \Delta(\cdot)$, $Sorted(m_i)$, being a decreasing ordering the m_i.
The author mentions in the paper and in the code that it is best to use it in combination with Cross Entropy loss, or to train the network with Cross Entropy first and then use the Lovász loss to finetune.

Tversky loss It [Salehi et al., 2017] adapts the Dice loss (Equation 2.6) in order to achieve a trade off between Precision (Equation 2.20) and Recall (Equation 2.21). Note that when $\alpha = \beta = 0.5$, the Tversky loss becomes Dice loss.

$$\mathrm{TL} = 1 - \frac{\sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c g_i^c}{\sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c g_i^c + \alpha \sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c (1 - g_i^c) + \beta \sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} (1 - p_i^c) g_i^c} \tag{2.12}$$

Focal Tversky loss (FTL) The Focal Tversky loss [Abraham and Khan, 2019] is proposed to improve Precision and Recall balance. The definition is as below, the γ in the paper is in 1,2,3. Note that when $\gamma = 1$, it becomes Tversky loss.

$$\mathrm{FTL} = (\mathrm{TL})^{\frac{1}{\gamma}} \tag{2.13}$$

Asymmetric similarity loss (ASL) The motivation for ASL loss [Hashemi et al., 2018] is also to better adjust the weights of FP and FN (and to achieve a better balance between Precision and Recall), for which a weighting parameter β is introduced, defined as follows:

$$\mathrm{ASL} = 1 - \frac{\sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c g_i^c}{\sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c g_i^c + \frac{\beta^2}{1+\beta^2} \sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} p_i^c (1 - g_i^c) + \frac{1}{1+\beta^2} \sum_c^{\mathrm{C}} \sum_{i=1}^{\mathrm{N}} (1 - p_i^c) g_i^c} \tag{2.14}$$

Note that when $\alpha + \beta = 1$, the ASL becomes Tversky loss.

Distance based loss

This type of loss functions is aiming to minimize the distance between predicted results and the ground truth.
Boundary loss (BL) This loss [Kervadec et al., 2019] using the integral framework to approximate the distance, which can avoid local differential computations involving boundary curve points. $Dist(\cdot)$ is the distance map same with Equation 2.26, Equation 2.27, and Equation 2.4. In Equation 2.15, it computes the mismatch re-

gions of the two boundaries.

$$\mathrm{BL} = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} [\mathrm{D}ist(1-g_i)(1-g_i) - (\mathrm{D}ist(g_i)-1)g_i]p_i \tag{2.15}$$

Hausdorff Distance loss (HDL) It comes from the HD evaluation metric (See Equation 2.26). Because the HD metrics can't be used as loss functions directly, so the authors [Karimi and Salcudean, 2019] utilize the distance map to approximate the distance.

$$\mathrm{HDL} = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} [(p_i - g_i) \odot (\mathrm{D}ist(g_i)^2 + \mathrm{D}ist(p_i)^2)] \tag{2.16}$$

One should note that both of these two distance-based loss functions are combined with region-based loss in order to keep stability as mentioned in their experiments. [Ribera et al., 2019] have the similar idea based on Hausdorff distance who proposed a loss function called "weighted Hausdorff distance" loss for object localization. Another loss function called "contour loss" that takes into account distance information via the distance map of the ground truth, has shown interesting smoothing effect in a 3D segmentation setting [Jia et al., 2018].

Hybrid loss

Another type of loss function is to combine one loss function with another loss function with a weighted value in order to dealing with the same issue, which is class unbalanced problem. In the work of [Trullo et al., 2017], they combined Cross Entropy loss and Weighted Cross Entropy loss. Three hybrid loss functions are listed below:
Dice+Cross Entropy [Taghanaki et al., 2019] this loss simply summarize the Cross Entropy loss and Dice loss together.

$$\mathrm{D}ice\mathrm{CE} = \mathrm{C}ross\mathrm{E}ntropyloss + \mathrm{D}iceloss \tag{2.17}$$

Dice+Focal loss it [Zhu et al., 2019] combines Dice loss with Focal loss.

$$\mathrm{D}iceFocal = \mathrm{D}iceloss + \mathrm{F}ocalloss \tag{2.18}$$

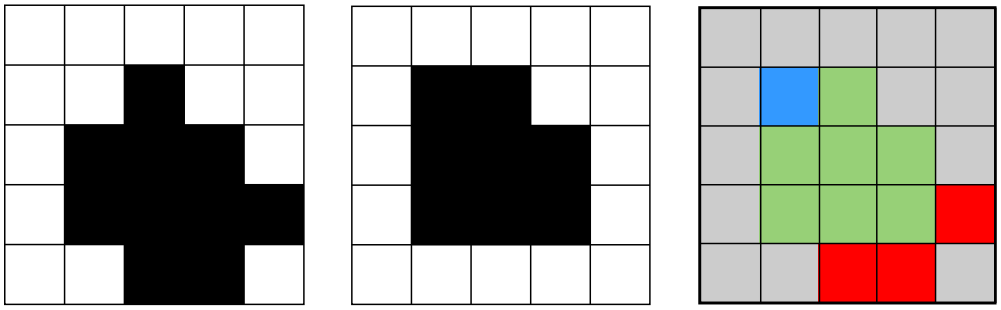
Exponential Logarithmic loss (ELL) [Wong et al., 2018] combines Dice loss and Cross Entropy loss in the exponential logarithmic way with respective weighting factors $w_{\mathrm{D}ice}$, w_{CE} and γ. In this exponential logarithmic Cross Entropy item, w is the

weight inside of the Cross Entropy loss to reduce the influences of more frequently seen labels.

$$\mathrm{ELL} = w_{\mathrm{D}ice}\mathrm{E}[(-ln(\mathrm{D}ice))^{\gamma}] + w_{\mathrm{CE}}\mathrm{E}[w(-ln(p_i))^{\gamma}] \tag{2.19}$$

2.1.4 Evaluation metrics in segmentation

Evaluating the segmentation results can reflect the strengths and weaknesses of a segmentation method. The following evaluation metrics are coefficients commonly used in medicine, and some are also statistical concepts often used in industrial production. We use Seg as segmentation results and GT as ground truth in the following mathematical expressions.



(a) Ground truth mask (b) Predicted binary mask (c) Overlap between masks

Figure 2.3 – A diagram of Ground truth image (a), predicted image (b) and the overlap between two masks (c). In (c), green pixels are TP, blue ones are FP, red ones are FN, grey ones are TN. The figure is adapted from [Taghanaki et al., 2021].

Region based metrics

Precision Precision (also called positive predictive value) is the proportion of true positives ($\mathrm{TP} = Seg \cdot \mathrm{GT}$) out of all detected positive instances including false positive ($\mathrm{FP} = Seg \cdot (1 - \mathrm{GT})$).

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2.20}$$

Sensitivity (Recall) Sensitivity (also called true positive rate) is the proportion of true positives out of all positive cases including false negatives ($\mathrm{FN} = (1 - Seg) \cdot \mathrm{GT}$).

$$Sensitivity = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{2.21}$$

Specificity Specificity (also called true negative rate) is the proportion of true nega-

tives (TN = $(1-Seg)\cdot(1-GT)$) out of all negative cases including false positives.

$$Specificity = \frac{TN}{TN+FP} \tag{2.22}$$

Dice coefficient The definition of Dice coefficient [Dice, 1945] is the proportion of overlap region over segmentation and ground truth in foreground part.

$$Dice = \frac{2Seg\cdot GT}{Seg+GT} \tag{2.23}$$

Jaccard coefficient The idea of Jaccard coefficient is similar with Dice coefficient, but a little different in mathematical formula.

$$Jaccard = \frac{Seg\cdot GT}{Seg+GT-Seg\cdot GT} \tag{2.24}$$

The relationship of Jaccard coefficient and Dice is as below [Taghanaki et al., 2021]:

$$Jaccard = \frac{Dice}{2-Dice} \tag{2.25}$$

Distance based metrics

Hausdorff Distance (HD) It is defined as the maximum surface distance ($Dist$) between the segmentation results and ground truth.

$$HD = Max(Max(Dist(Seg,GT)),Max(Dist(GT,Seg))) \tag{2.26}$$

Average symmetric surface distance (ASSD) It computes the average surface distance between the segmentation results and ground truth.

$$ASSD = Mean(Mean(Dist(Seg,GT)),Mean(Dist(GT,Seg))) \tag{2.27}$$

2.2 Direct biomarker estimation methods

In the last section (Section 2.1), we introduce image segmentation methods based on traditional algorithms and deep learning models. In fact, in medical image segmentation, in most cases, the segmentation result is only an intermediate step. This is because segmented areas are designed to quantify geometric factors such as perimeter, area or volume, which are then further translated into some sort of

biomarker in clinical medicine. Therefore, there have been researches that try to skip segmentation, and focus on direct estimation the biomarker. In the following sections, we will introduce the direct prediction objects as well as methods.

2.2.1 Traditional machine learning methods on direct estimation

Before deep learning methods coming up, early machine learning techniques are usually used in direct estimation of indices on different study targets. Basically these methods need to manually extract the features from input images, then feeding them into regressors to directly estimate the values given the regression loss and ground truth. Table 2.1 summarizes the traditional machine learning methods on different applications. Note that the data they use are 2D cardiac images, which means that they predict the area first then add the areas slice by slice to form the volume. Generally, traditional machine learning methods need to building hand-crafted feature through statistical learning methods like Bhattacharyya coefficients [Afshin et al., 2012, Zhen et al., 2015b], histogram of oriented gradients (HoG) [Zhen et al., 2014], supervised descriptor learning (SDL) [Zhen et al., 2015a] etc at first. Then these features are sent into different models such as artificial neural networks (ANN) [Afshin et al., 2012], support vector machine (SVM) [Afshin et al., 2013, Sun et al., 2017], Bayesian model [Wang et al., 2014, Zhen et al., 2015b] and random forest [Zhen et al., 2014, Zhen et al., 2015b, Zhen et al., 2015a, Zhen et al., 2016a, Li et al., 2017] to regress the estimated results by regressional objective function like mean absolute error or mean square error. Clustering method can also be used in biomarker estimation, in [Ivanov et al., 2019], the authors address with LV volume estimation problem in 3 steps: 1.Locate LV; 2. Identify ED, ES, calculate area of LV by performing clustering algorithm so that the largest cluster of the image is considered to be the left ventricle; 3. Compute volume of LV. There are other machine learning algorithms such as manifold learning [Wang et al., 2015, Sun et al., 2017, Tan et al., 2020], multi-output and multi-target regression [Zhen et al., 2016b, Zhen et al., 2017a, Zhen et al., 2017b] and regularization method [Gu et al., 2018].

2.2.2 Deep learning methods on direct estimation

Compared to early machine learning methods, which are featured with multi-stage learning. The deep learning methods often come with end-to-end learning, more

Table 2.1 – Traditional machine learning methods for biomarker estimation from cardiac images.

Reference	Estimation object	Method	Data
[Afshin et al., 2012]	EF	Manual feature extraction+ANN	2D Cardiac MRI data
[Afshin et al., 2013]	volume of LV	Manual feature extraction+SVM	2D Cardiac MRI data
[Wang et al., 2014]	volume of bi-LV	Manual feature extraction+Bayesian model	2D Cardiac MRI data
[Zhen et al., 2014]	volume of bi-LV	Manual feature extraction+Random forest	2D Cardiac MRI data
[Wang et al., 2015]	clinical variables	Manifold learning	2D Cardiac MRI data
[Zhen et al., 2017b]	volume of four chamber	Multi-ouput and multi-target regression	2D Cardiac MRI CT data
[Ivanov et al., 2019]	volume of LV	Clustering method	2D Cardiac MRI data

importantly, they can automatically learn features from images by various CNN architectures. Therefore, researchers are dedicated to designing high-efficiency networks to specific applications. We summarize the deep learning methods on direct quantification of different applications into Table 2.2 from its origins to recent research findings, which is mainly in the area of medical imaging. From this table we can know that the application range is rich, and the deep learning methods are various based on different demands. In this section, we broadly classify the studies of deep learning-based direct prediction of biomarker from medical images into the following categories and will describe it in detail in the following content.

- Multi-scale learning
- Multi-task learning
- Attention mechanism
- Segmentation and reconstruction based regression
- Hybrid statistical learning with deep learning
- Temporal and spatial networks

Multi-scale learning

The so-called multi-scale is actually sampling the signals/images at different scales, and usually at different scales we can observe different features to accomplish different tasks. Then the intention of multi-scale learning is to enlarge the reception field in the networks. The specific network structure can be classified as follows: (1) Multi-scale input. (2) Multi-scale feature fusion. (3) Multi-scale model fusion. (4) Combination of the above methods.

Multi-channel fusion Multi-channel or multi-view fusion techniques is to fuse images of different modals or positions in order to get sufficient features through CNN

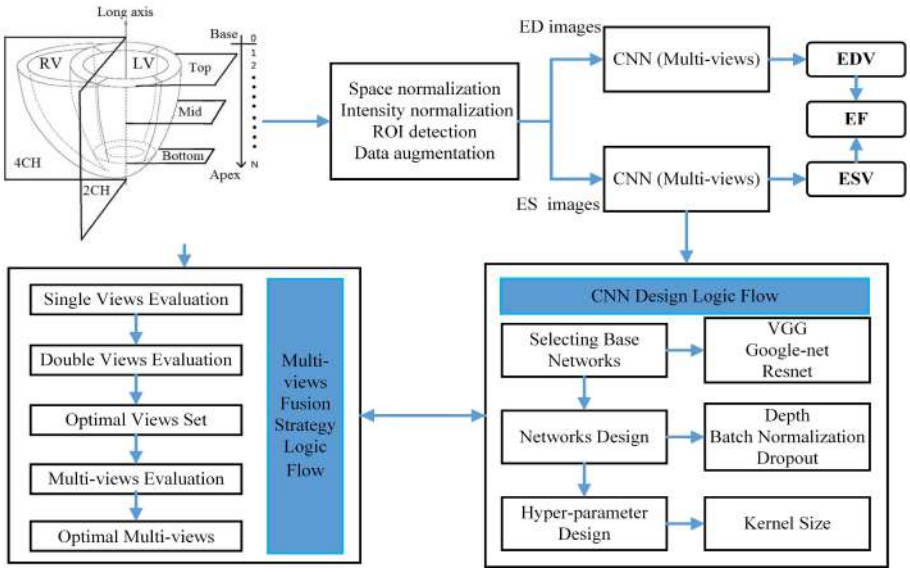


Figure 2.4 – Multi view/channel fusion strategy, the input is cardiac top slice, middle slice and bottom slice, the figure is obtained from [Luo et al., 2017].

layers. However, there is no accurate solution to point out which combination is the best fusion. It depends on the quality of data, actually. Therefore, [Luo et al., 2017] propose multi-view fusion strategy through quite a few experiments. By the end, they chose the <Top, Mid, 2CH> slice as the final multi-views fusion strategy because they have the smallest RMSE value, see Figure 2.4.

Multi-feature fusion [Zhen et al., 2016a] utilize multi-scale feature fusion strategy by applying different sizes of CNN kernels in networks to directly predict the biventricular volume, see Figure 2.5.

Multi-model fusion Another idea [Zhang et al., 2019, Zhang et al., 2020a] utilise isolated 3D convolution networks in each view then fuse each corresponding regression models for extracting multi-view features of coronary artery. [Li et al., 2020] proves that feature fusion (through 3 cascaded modules,the cardiac cycle extraction module, the motion feature extraction module, and the fully connected regression module) has a positive effect on the direct estimation of the LV. [Luo et al., 2020b] not only fuses different views of input medical image slices, but also fuses different models to dynamically rectify the prediction results.

Multi-task learning

Multi-task learning, which literally means different tasks (models) are integrated in one networks and they are learning simultaneously. The benefit of multi-task

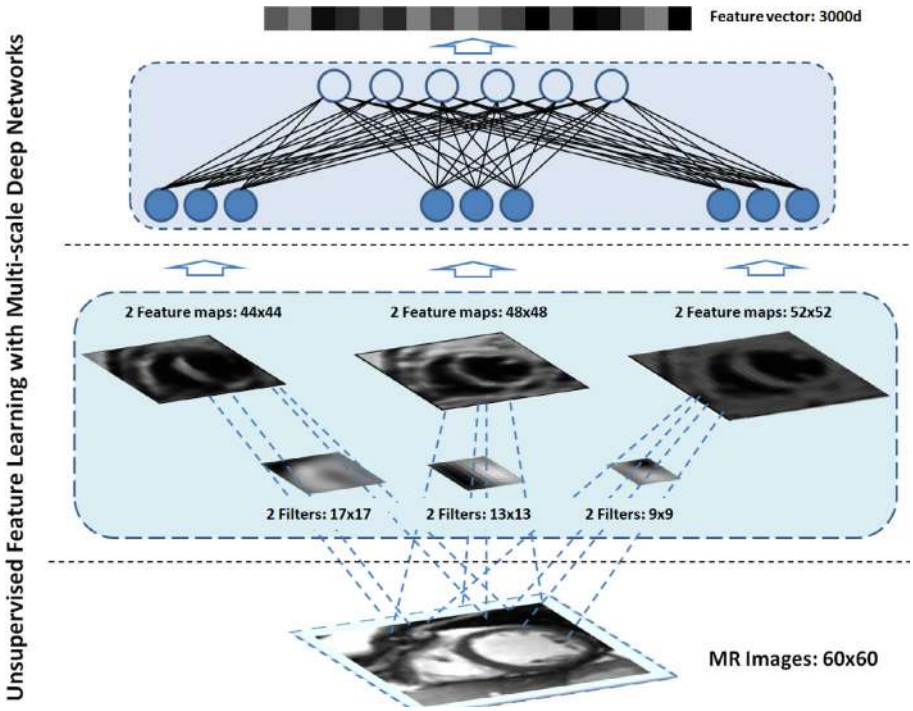


Figure 2.5 – Multi-feature fusion strategy, the input is cardiac images, the feature maps are in different size because of different kernel/filter size, the figure is obtained from [Zhen et al., 2016a].

learning is that the weights are shared in different tasks, so that each task can learn sufficient and complementary features from input medical data and corresponding loss functions. In [Xue et al., 2017b, Xue et al., 2018, Du et al., 2018], the authors combine the CNN and RNN architectures to estimate the indices of cardiac and two phases ED and ES (see Figure 2.6). In [Dangi et al., 2018, Chen et al., 2020b], the authors combine the U-Net [Ronneberger et al., 2015] and regression layer as multiple output layers, so that the regression layer can not only learn the input image but also learn from segmented images, and this two parts of weights are shared. More advanced, [Luo et al., 2020a] combines the segmentation model (FCN [Long et al., 2015]) and regression model, they also build a mutual authentication bridge between this two model through a loss function to minimize the difference between two output modules. In [Xu et al., 2018], the authors combine the generator and discriminator network with regression layer to achieve segmentation and direct estimation of multiple cardiac indices (Myocardial Infarction, MI). In [Ge et al., 2019a], the authors combine segmentation models with RNN models as well as regression models. In [Huang et al., 2021b, Liu et al., 2020], there is one

model with two output branches, they are regression layer for direct indices prediction and classification layer, respectively. In [Vesal et al., 2020], their model perform 3 tasks simultaneously, respectively are segmentation, regression and classification. The multi task learning can also happen in different image modalities. That is to say, the parameters from the two networks learned from the source modality (MRI) are shared with the target modality (CT) [Yu et al., 2021].

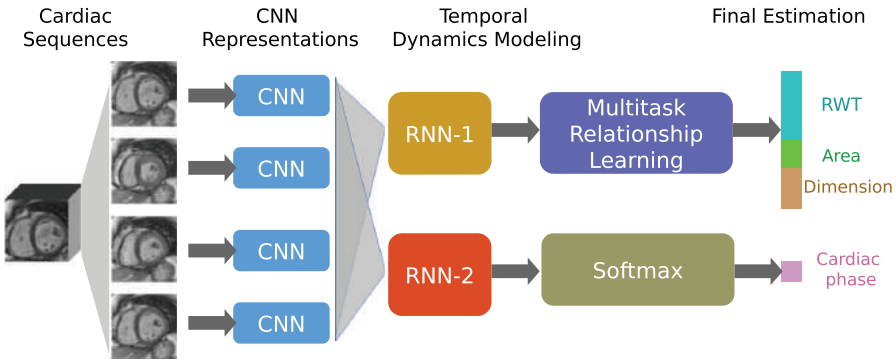


Figure 2.6 – Multi-task learning with different neural networks, the figure is obtained from [Xue et al., 2018]

Segmentation and reconstruction based regression

Due to the original images have unclear features and noise, it's natural to think up of using segmentation or reconstruction results to estimate the indices from medical images. There are several indices estimation methods are based on the segmentation results [Liao et al., 2017, Du et al., 2018, Wang et al., 2019, Liu et al., 2018, Tao et al., 2019, Pereira et al., 2020], they first utilise segmentation neural networks (e.g. U-Net [Ronneberger et al., 2015]) to obtain binary or multi-class segmentation results, then it will be is easier to estimate indices through regression CNN. The same idea was used in [Gessert and Schlaefer, 2019], but the models (2D and 3D CNN) are initialized with pretrained weights from ImageNet [Deng et al., 2009]. This operation require two kinds of labels, one is ground truth contour of to-be-segmented images, the other is ground truth of indices. In [Xue et al., 2017a], they set up a encoder (convolution) and decoder (deconvolution) networks to reconstruct the input medical data, then the multiple indices of cardiac are estimated from the reconstructed images, see Figure 2.7.

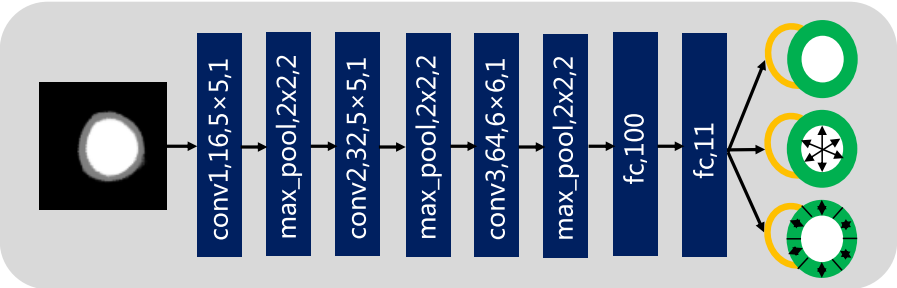


Figure 2.7 – Segmentation results as input training data, the outputs are indices of cardiac, the figure is obtained from [Wang et al., 2019].

Hybrid statistical learning with deep learning

Although deep learning-based methods have excellent performance, the previous statistical or machine learning methods also performed well. Thus, some researchers combine machine learning and deep learning methods to get a better results. In [Zhen et al., 2016a], they combine CNNs with random forest to estimate biventricles, see Figure 2.8. In kidney volume estimation, [Hussain et al., 2016] also combine CNNs with dual regression forests.

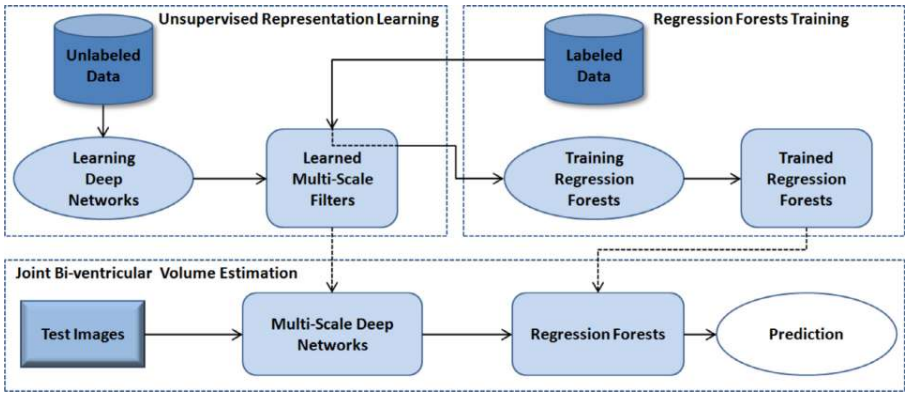


Figure 2.8 – The flowchart of the feature learning and random forest regression, the figure is obtained from [Zhen et al., 2016a].

Temporal and spatial networks

In cardiovascular disease diagnose, Ejection Fraction (EF, see Equation 1.1) is a common metric, of which doctors need to know the volume in ED and ES two phases, one way for identifying ED or ES is recognizing them by experienced doctors' eyes. For instance, in the ACDC dataset [Bernard et al., 2018], the ED

and ES are already labeled by experts. However, with the number of images increasing, manual recognizing two phases is labouring. Thus, researchers come up with automatic identifying ED and ES by using temporal and spatial networks [Xue et al., 2017c, Luo et al., 2019, Ge et al., 2019b]. Specifically, the networks [Xue et al., 2017c] are composed of Recurrent Neural Networks (RNN) or Long Short-term Memory (LSTM) and CNNs. Fig. 2.9 is one temporal regression CNN networks. The input data usually is 4D MRI or CT data, that is 3D image as well as different frames of the process of a heart beat. The networks [Luo et al., 2019] consists of a shared weights framework but has two outputs, one is the temporal ordered value (judging ED or ES), the other one is estimated volume of LV, each task has a corresponding loss function, this is also counted as multitask learning. A more flexible networks [Liu et al., 2021b] that can support any number of frame input so that it predicts the left ventricle indices frame-by-frame through encoder (gated recurrent unit, GRU [Chung et al., 2014]) and decoder (GRU with attention mechanism).

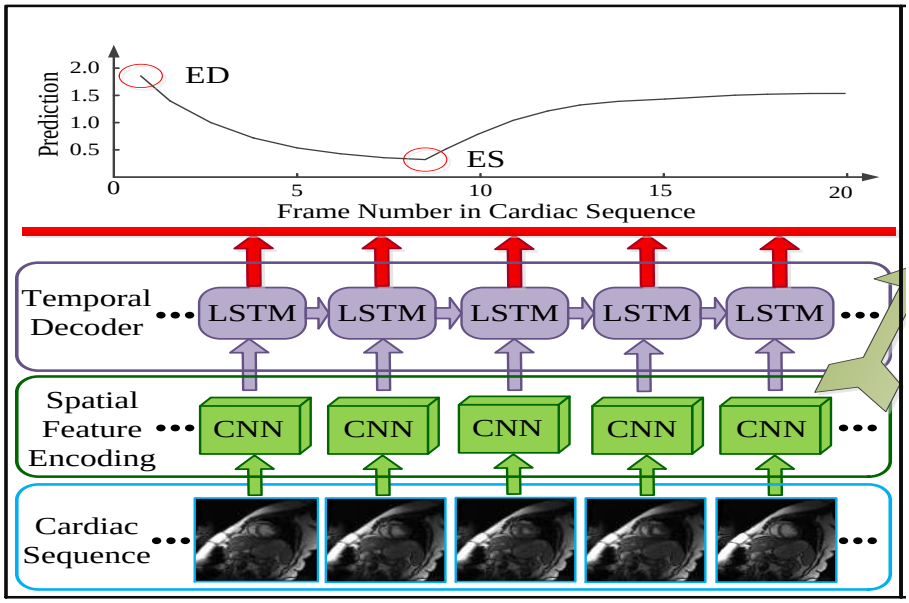


Figure 2.9 – Temporal regression CNN networks, which can predict ED or ES phase from continuous sequence of cardiac frames, the figure is obtained from [Kong et al., 2016].

Attention mechanism

The attention mechanism is an additional option to amplify the learned features so that the model can well predict the values. It origins from natural language processing, then popular in computer vision. There are several kinds

of attention methods: content-base attention [Graves et al., 2014], additive attention [Bahdanau et al., 2014], [Luong et al., 2015] proposed location base attention, general attention and dot-product attention, scaled dot-product attention [Vaswani et al., 2017]. Now quite a few literature start to utilize attention mechanism in different biomarker prediction tasks [Pang et al., 2018, Pang et al., 2019, Ge et al., 2019a, Liu et al., 2021b].

Medical image datasets used for direct estimation

We list the public medical image datasets that have been used for direct estimation, see Table 2.3. These challenges are oriented to segmentation tasks at first, and mainly focused on cardiac images. The other dataset of different organs or tissues mentioned in this survey were not publicly accessible. Due to these original datasets are not suitable for directly use in different methods, for instance, the amount, the size, the dimension, etc. Thus it's necessary to apply preprocessing before using them on specific tasks and solutions. In general, data augmentation, normalization, resizing and cropping, Region of Interest (ROI) identification as well as slice selection in 3D data are common operations.

2.2.3 Evaluation metrics in regression

Mean absolute error (MAE) measures the error from predicted value and ground true value, see Formula 2.28, where N is the number of total samples, x_i and y_i are estimated and ground truth values.

$$\text{MAE} = \frac{1}{\text{N}} \sum_{i=1}^{\text{N}} |x_i - y_i| \tag{2.28}$$

Percentage MAE Usually only the MAE is not enough, one should also know the ratio of the prediction error compared to the true result. Thus the percentage MAE (PMAE) is defined by Formula 2.29, which is also called error rate.

$$\text{Error rate} = \frac{\sum_{i=1}^{\text{N}} |x_i - y_i|}{y_i * \text{N}} \tag{2.29}$$

Correlation coefficient Another common used metrics is correlation coefficient r, see Formula 2.30, $\overline{x}$ and $\overline{y}$ are the mean values estimated and ground truth values.

$$r = \frac{\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \overline{x})^2 (y_i - \overline{y})^2}} \tag{2.30}$$

Root mean square error (RMSE) is also used in comparing the estimated value and ground truth value.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - y_i)^2}{N}} \tag{2.31}$$

Statistical significance tests. These tests can be performed when the scores obtained by two methods are close, in order to decide the wheter the difference between them is significant or not. The t-test helps to determine whether the difference between two set of values, based on the average, is significant. The paired Wilcoxon test is a non-parametric alternative to paired t-test used to compare paired data. It's used when the data are not normally distributed.

Table 2.3 – Public medical image datasets used in direct estimation.

Datasets	**Year**	**Number of subjects**		**Ground truth**	**Type**
		training set	**test set**		
HC18 [1]	2018	999	335	head circumference	US
Sunnybrook [2]	2009	45	-	LV, MYO, Pathology	MRI
LVSC [3]	2011	100	100	LV, Pathology	MRI
MICCAI RV [4]	2012	16	32	RV	MRI
Kaggle [5]	2015	500	300	Cardiac Volumes	MRI
ACDC [6]	2017	100	50	LV, RV, MYO, Pathology	MRI
LVQUAN18 [7]	2018	145	30	Cardiac indices	MRI
LVQUAN19 [8]	2019	56	30	Cardiac indices	MRI

[1] `https://hc18.grand-challenge.org/`

[2] `http://smial.sri.utoronto.ca/LVChallenge/Home.html`

[3] `www.cardiacatlas.org/challenges/lv-segmentation-challenge/`

[4] `http://rvsc.projets.litislab.fr/`

[5] `www.kaggle.com/c/second-annual-data-science-bowl`

[6] `https://www.creatis.insa-lyon.fr/Challenge/acdc/index.html`

[7] `https://lvquan18.github.io/`

[8] `https://lvquan19.github.io/`

2.2.4 Perspectives

Through this survey one can find that the direct quantification methods have promising prospects on medical images, such as in the fields of head circumference prediction, spine Cobb angle prediction, kidney disease diagnose by volume prediction, and cardiovascular disease diagnose by some medical indices prediction. Especially in cardiac problem, great efforts are put by plenty of researchers through various traditional machine learning or deep learning based methods. However, due to the quality and types of medical images as well as deep learning models are various, thus the author have the following perspectives.

Data type: Currently, the medical images cover MRI, CT, Ultrasound, X-Rays, they are 2D or 3D formats. But in this survey, almost all the studies convert the 3D images into 2D slices. For example, in cardiac indices or volume estimation problem, they input 2D slice or 2D+Time slice, but not the whole 3D scanned data, that is to say, they predict the area first then sum up the area of each slice.

Preprocessing: Because of the complexity of dataset, data preprocessing is necessary in every research. If the data amount is limited, then data augmentation should be performed before or during model training process. Moreover, data resizing, cropping and normalization are common operation in deep learning. If the target is small in the whole image, then ROI detection can largely reduce processing time and improve efficiency, for instance in cardiac data, it just needs to focus on two or four chambers but not the other parts of body. Also, certain slice selection can be done depend on specific demand which may improve the performance because of clear features. However, because this step (data and preprocessing strategy) in each experiment is very different from the other papers, which leads to the results hard to be compared.

Methods: We can clearly see that the deep learning methods in recent years dominate in various applications. In deep learning methods, they are divided into several sub branches such as multi-scale learning, multi-task learning, attention mechanism, combining traditional machine learning with deep learning methods, etc. Therefore, when the data is appropriate and the model is designed reasonably, satisfactory results can generally be obtained. However, interpretability is quite important in medical imaging analysis and disease diagnose. If human being can understand and trust the explainable unseenable deep learning black box, then this type of deep learning methods are reliable to applied in clinical medical applications and become a right-hand man.

In the future, in addition to the existed research objects, the author believes that there will be more and more clinical medical indices that can be directly estimated by various efficient deep learning methods with less errors. So that they can assist doctors in their judgment and decision-making. For example, in oncology, the anthropometric parameters like muscle body mass (MBM), fat body mass (FBM), lean body mass (LBM), visceral adipose tissue (VAT), and subcutaneous adipose tissue (SAT) etc [Decazes et al., 2019] are important indices to evaluate a human's health state, which are potential possible applications.

Table 2.2 – Deep learning methods on direct quantification of different applications.

Reference	Application	Method
CNN+regression (Reg) model		
[Riegler et al., 2013]	Head pose estimation	Hough Forests with CNNs+Reg
[Luo et al., 2016]	LV volume estimation	8-layer CNN models with MSE loss
[Chen et al., 2016]	Mitosis counting for breast cancer	CNN+Reg layer
[Zhang et al., 2020b]	Head Circumference prediction	Pretrainded CNN+Reg
Multi-scale		
[Zhen et al., 2016a]	Bi-ventricle volume estimation	Multi-scale kernels
[Luo et al., 2017]	LV volume estimation	Multi-view input(2CH,top+mid)
[Zhang et al., 2020a]	Quantify Coronary Artery Stenosis	Multi-view parallel feature fusion
[Li et al., 2020]	LV volume estimation	Cascaded feature fusion
[Luo et al., 2020b]	Bi-ventricle volume estimation	Multi-view input and feature fusion
Multi-task		
[Xue et al., 2018]	Quantify all LV indices(11)	CNN+RNN
[Dangi et al., 2018]	LV Seg, cardiac indices estimation	U-Net+Reg
[Xu et al., 2018]	MI Seg and quantification	Multi-task GAN, Generator: Reg+Seg; Discriminator:Bi-LSTM networks
[Luo et al., 2020a]	Bi-ventricle volume estimation	Seg and Reg module and mutual authentication module between them
[Liu et al., 2020]	EF estimation	Classification+Regression
[Vesal et al., 2020]	LV indices quantification	Classification+Segmentation+Regression
[Yu et al., 2021]	LV indices quantification	Shared parameters between MRI and CT.
[Zhao et al., 2021]	Carotid artery indices estimation	Cell detection, segmentation, classification
Attention mechanism		
[Pang et al., 2019]	Multiple indices of spine estimation	Cascade feature amplifier network
[Ge et al., 2019a]	LV indices quantification	Attention junction from Seg to Quantify
[Liu et al., 2021b]	LV indices quantification	Attention integrated into decoder
Segmentation (Seg)/reconstruction based regression		
[Du et al., 2018]	estimate the EF	Seg results as regression CNN input
[Liu et al., 2018]	LV volume estimation	Seg module(U-Net) and Regression CNN
[Wang et al., 2019]	LV indices quantification	Seg module and Regression module
[Pereira et al., 2020]	LV indices quantification	Seg results and original images as regression model input
[Gessert and Schlaefer, 2019]	LV indices quantification	Seg module and pretrained Reg CNN
[Xue et al., 2017a]	LV indices quantification	Reconstruction model+Reg CNN
Statistical mixed with deep learning		
[Zhen et al., 2016a]	Bi-ventricular volume estimation	CNN+Random forest model
[Hussain et al., 2016]	Kidney volume estimation	CNN+Random forest model
Temporal and spatial networks		
[Xue et al., 2017c]	LV indices quantification	RNN/LSTM model+Reg CNN model
[Luo et al., 2019]	ED and ES prediction LV volume estimation	Two parameter-shared networks Ranking model+Estimation model

2.3 Explainable Artificial Intelligence

In this section, we will present the explanation methods and tools that have been developed for deep learning models for vision applications, as well as how to evaluate these methods.

2.3.1 Explanation methods

Saliency maps

A saliency map is supposed to highlight the pixels that most contributed to the network's decision, with regards to one specific image; it is also called pixel attribution [Molnar, 2019]. Figure 2.10 is an example of saliency map based on a image classification task from CIFAR-10 (Canadian Institute For Advanced Research) dataset [Krizhevsky et al., 2009]. The saliency map is a main carrier to visually explain the deep learning models. In the following contents, we will introduce several types of explanation methods that can generate saliency maps.

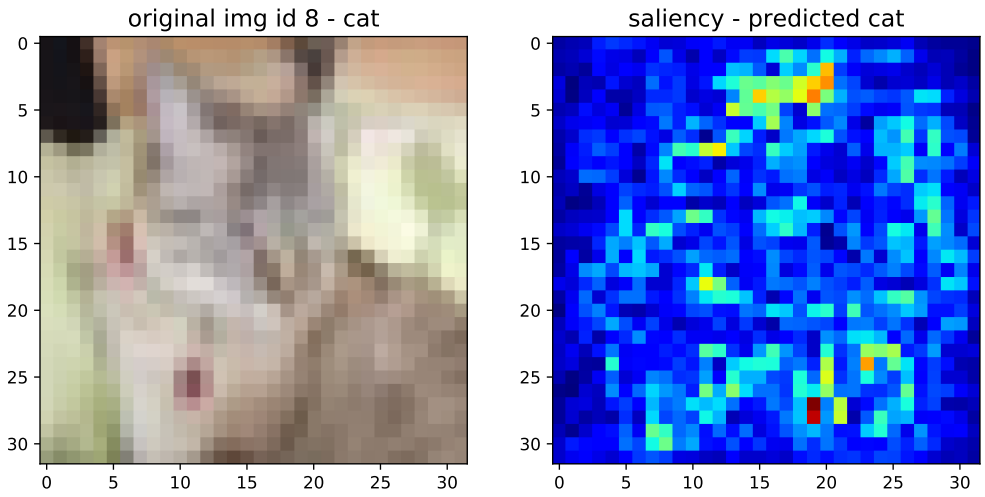


Figure 2.10 – A test image (a cat) from CIFAR dataset and the gradient-based saliency map of the test image predicted on a customize CNN model. A saliency map in which pixels are colored by their contribution to the classification.

One can also extract the feature maps in the intermediate of the model in order to observe what each layer has learned. Figure 2.11 shows feature maps of different layers. The output of each feature map is the weight value of each neuron.

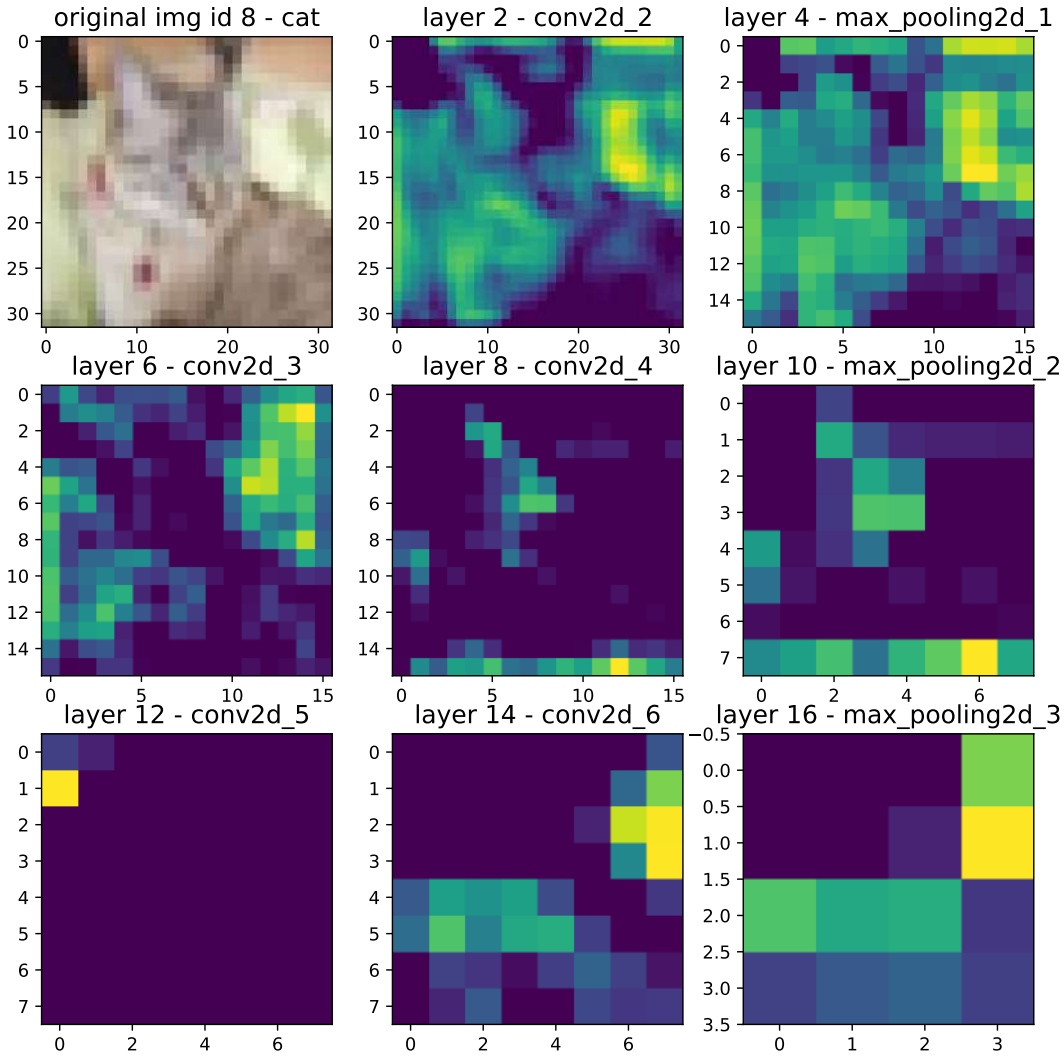


Figure 2.11 – A test image (a cat) from CIFAR dataset and feature maps of different layers of a customize CNN model.

Gradient based methods

Gradient [Simonyan et al., 2013]: The gradient of the output neuron with respect to the input. The input here is an image that can be represented $\{x_1, x_2, ...x_N\}$, the corresponding output neuron is a predicted value y. Putting Δx to each pixel of the image to see the change of y, which is $y + \Delta y$, then computing $\frac{\Delta y}{\Delta x}$. Then the saliency map can be generated by calculating the impact of each input pixel to the output value (gradient). The area with higher brightness represents the greater the influence of this pixel on the prediction result.

Grad-CAM [Selvaraju et al., 2017]: Gradient-weighted Class Activation Mapping

uses the gradients of any target concept or certain class, flowing into the final convolutional layer to produce a coarse localization map highlighting the important regions in the image for predicting the concept. It is a generalization of CAM [Zhou et al., 2016] method.

SmoothGrad [Smilkov et al., 2017]: it averages the gradient over number of inputs with added noise. See Equation 2.32, where n is the number of random samples in a neighborhood of an input x, $\mathrm{N}(0,\sigma^2)$ represents Gaussian noise. G is gradient.

$$\hat{\mathrm{G}}(x) = \frac{1}{n}\sum_{1}^{n}\mathrm{G}(x+\mathrm{N}(o,\sigma^2)) \tag{2.32}$$

Input*Gradient [Shrikumar et al., 2016]: It multiplies the input image with the gradient value.

Integrated Gradients [Sundararajan et al., 2017]: It integrates the gradient along a path from the input to a reference.

$$\mathrm{IG}(input,ref) = (input-ref)*\int_{0}^{1}\Delta\mathrm{G}(\alpha*input+(1-\alpha)*ref)d\alpha \tag{2.33}$$

DeConvNet [Zeiler and Fergus, 2014]: it performs the mapping with a Deconvolutional Network, which including unpooling, recitfication and filtering in the unsupervised way. It applies a ReLU in the gradient computation instead of the gradient of a ReLU.

Guided BackProp [Springenberg et al., 2015]: it applies a ReLU in the gradient computation additionally to the gradient of a ReLU. In other words, it combines the methods Gradient and DeConvNet. See Fig 2.12, in which it compares the different methods among Gradient, DeConvNet and Guided Backpropagation.

Attribution based methods

Class Activation Mapping (CAM) [Zhou et al., 2016]: the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps (CAMs). The CAM highlights the class-specific discriminative regions. Specifically, they utilise global average pooling to produce the spatial average of the feature map of each unit at the last convolutional layer. A weighted sum of these values is used to generate the final output. Similarly, a weighted sum of the feature maps of the last convolutional layer is computed to obtain class activation maps.

DeepTaylor [Montavon et al., 2017]: It computes for each neuron a rootpoint $((\widetilde{x}_i)_i)$, that is close to the input, but which's output value is 0, and uses this dif-

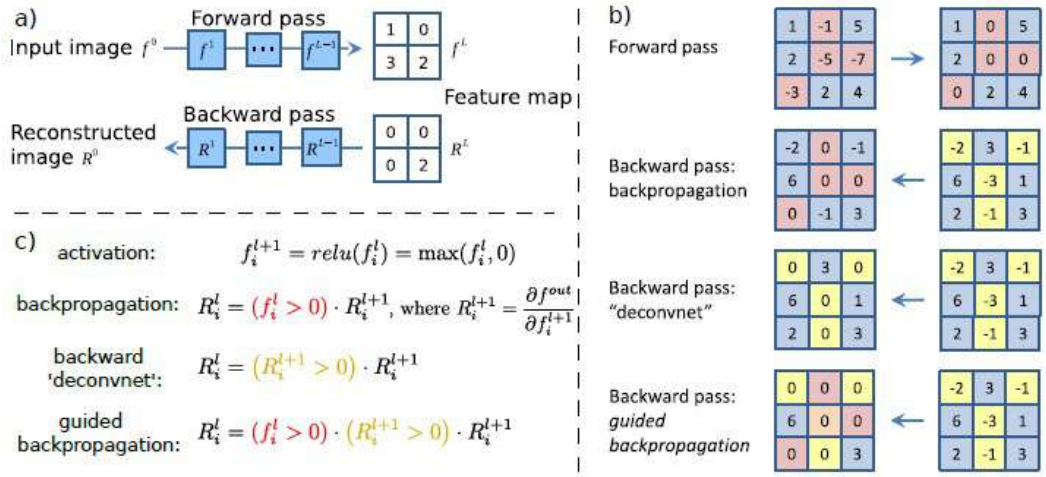


Figure 2.12 – Comparison of gradients, deconvnet, guided backpropagation methods (Figure is adapted from [Springenberg et al., 2015].

ference to estimate the attribution of each neuron recursively. Decomposition is continuous everywhere in the input domain: Two nearby points in the input space always have a similar explanation (provided that the function is continuous). Furthermore, the magnitude of the decomposition (size of the arrow) is proportional to the function value at a given point in space. Whereas sensitivity analysis measures a local effect.

$$[x_f]_j = \sum_i c_j \frac{\partial x_j}{\partial x_i}\Big|_{(x_i)_i=(\tilde{x}_i)_i} \cdot (x_i - \tilde{x}_i) \tag{2.34}$$

Layer Relevance Propagation (LRP) [Bach et al., 2015]: It attributes recursively to each neuron's input relevance proportional to its contribution of the neuron output. The magnitude of the contribution of each pixel or intermediate neuron is called "relevance" values R.

$$\mathrm{R}_j = \sum_k \frac{a_j w_{jk}}{\sum_{0,j} a_j w_{jk}} \mathrm{R}_k$$

Here, j and k are two neurons of any consecutive layers. We already know the relevance R in the output layer, so we'll start from there and use this formula iteratively to calculate R for every neuron of the previous layer. a denotes the activation of the respective neuron, and w is the weight between the two neurons.

Perturbation based methods

Local Interpretable Model-Agnostic Explanations (LIME) [Ribeiro et al., 2016]: The principle of LIME method is to perturb the input and see how the predictions change. This turns out to be a benefit in terms of interpretability, because it can perturb the input by changing components that make sense to humans (e.g., words or parts of an image), even if the model is using much more complicated components as features (e.g., word embeddings).
Meaningful Perturbation [Fong and Vedaldi, 2017]: Similar to LIME method, in the work of meaningful perturbation, they delete some areas of the image and observe the influence to model's prediction. Their method is faster to converge than LIME.

Miscellaneous

Anchors [Ribeiro et al., 2018]: It's a method of rule-based, model-agnostic explanations called anchors, designed to exhibit both these properties. Anchors highlight the part of the input that is sufficient for the classifier to make the prediction, making them intuitive and easy to understand.
SHapley Additive exPlanations (SHAP) [Castro et al., 2009]: SHAP assigns each feature an importance value for a particular prediction. It calculates the marginal contribution of features to the model output, and then explain the "black box model" from both global and local levels. SHAP constructs an additive explanatory model, and all features are regarded as "contributors".
RISE [Petsiuk et al., 2018]:It estimates importance empirically by probing the model with randomly masked versions of the input image and obtaining the corresponding outputs.
DeepDream [Mordvintsev et al., 2015]: The idea in DeepDream is to choose a layer (or layers) and maximize the "loss" in a way that the image increasingly "excites" the layers. The complexity of the features incorporated depends on layers chosen by users, i.e, lower layers produce strokes or simple patterns, while deeper layers give sophisticated features in images, or even whole objects.
Testing with Concept Activation Vector (TCAV) [Kim et al., 2018]: Unlike saliency explaining a single example, TCAV tries to explain a concept in terms of human-friendly and find the corresponding visual pattern.
So far, most of explanation methods are based on post-hoc methods. In this section, we list and introduce classic explanation methods. In the end, Table 2.4 summarizes

the main high cited explanation methods that are used in different areas.

2.3.2 Libraries and tools of XAI

Table 2.5 – Explanation tools through different platforms.

Tools	Category	Tools	Category
Heatmapping[3]	web	CNN Explainer[4]	web
Explainable AI Demos[5]	web	A Neural Network Playground[6]	web
Summit[7]	web	NeuralDivergence[8]	web
SCIN[9]	web	Neuroscope	free software
LUCID[10]	library	Keras-vis[11]	library
DeepExplain[12]	library	iNNvestigate[13]	library
TensorFlow Graph Visualizer[14]	library	tf-explain[15]	library
TorchRay[16]	library	Captum[17]	library
What-If Tool[18]	library	SHAP[19]	library
Interpret[20]	library	Eli5[21]	library
Skater[22]	library	GANDissect[23]	library
Yellowbrick[24]	library	AIF360[25]	library
Alibi Explain[26]	library	AIX360[27]	library
Explainable AI[28]	commercial	exAID[29]	commercial
H2o[30]	commercial	DASL[31]	commercial
SCOPA[32]	commercial		

3 http://www.heatmapping.org/
4 https://poloclub.github.io/cnn-explainer/
5 https://lrpserver.hhi.fraunhofer.de/
6 https://playground.tensorflow.org/
7 https://fredhohman.com/summit/
8 http://haekyu.com/neural-divergence/
9 https://www.dfki.de/skincare/classify.html
10 https://github.com/tensorflow/lucid
11 https://raghakot.github.io/keras-vis/
12 https://github.com/marcoancona/DeepExplain
13 https://github.com/albermax/innvestigate
14 https://www.tensorflow.org/tensorboard/graphs
15 https://tf-explain.readthedocs.io/en/latest/
16 https://github.com/facebookresearch/TorchRay
17 https://captum.ai/
18 https://pair-code.github.io/what-if-tool/
19 https://github.com/slundberg/shap
20 https://github.com/slundberg/shap
21 https://github.com/TeamHG-Memex/eli5
22 https://github.com/oracle/Skater
23 https://github.com/CSAILVision/GANDissect
24 https://github.com/DistrictDataLabs/yellowbrick
25 https://github.com/Trusted-AI/AIF360
26 https://github.com/SeldonIO/alibi
27 https://github.com/Trusted-AI/AIX360
28 https://cloud.google.com/explainable-ai
29 https://exaid.kl.dfki.de/
30 https://www.h2o.ai/products-dai-mli/
31 https://www.decodedhealth.com/
32 https://datalanguage.com/scopa-scalable-explainable-ai

Plenty of XAI tools have been developed to users in specific areas. See Table 2.5, some of them are website of tutorials or visualization of interaction with user input images (or model parameters). There are many Python libraries about various explanation methods that can directly be installed locally and used by users. More sophisticatedly, there have been commercial explanation tools. This phenomena

indicates that the explainable AI is everywhere and of much necessity in different areas and quite a lot researcher are dedicated in making the deep learning technology understandable and trustable.

2.3.3 Applications of explainable AI

According to the research of the existing literature, the XAI technology are used in various media or data such as text, image, graph, audio, electrocardiogram (ECG) etc. See Table 2.6. Moreover, explanation techniques would show up in each field such as medical, transportation, finance etc as long as they apply deep learning methods in these solutions.

Table 2.6 – Applications and medium of explanation methods

Media	**Applications**
Text	NLP [Liu et al., 2019], Finance, Social media, Sales, Human resources, Energy
Image	Medical images[Tjoa and Guan, 2021], Natural images
Graph	GNN explainer [Ying et al., 2019]
Audio	Speech recognition [Becker et al., 2018]
ECG	DeepExplain ECG [Raghunath et al., 2020]

2.3.4 Evaluation of explanation methods

With widely used of explanation techniques on deep learning architectures, the evaluation standards of XAI should be established. In [Arya et al., 2019], they state the evaluations of XAI should be of: Competence, Fairness, Safety, Usability,Human-AI collaboration, Accountability, Privacy, in [Goebel et al., 2018], they propose: Comprehensibility, Succinctness, Actionability, Reusability, Accuracy, Completeness. In [Samek et al., 2021], they propose Faithfulness/Sufficiency, Human Interpretability, Applicability and Runtime. From these evaluation critics We can see that they are still abstract and not so mature to perform. In this section, we list 3 executable methods, they are:

- Sanity checks [Adebayo et al., 2018b]
- Area over Perturbation Curve [Samek et al., 2016]
- Input variant [Kindermans et al., 2019]

Sanity checks

In the method of sanity checks [Adebayo et al., 2018b], they propose an idea to explain a deep neural networks model, that is to perform sanity checks on a certain model in both input data and model parameter two aspects to see the change of saliency maps.

The model parameter randomization test: Comparing the trained and untrained two models, if there is no difference in the saliency maps, it indicates that the saliency map method is not sensitive to the inspection of the model parameters and is not helpful.

The data randomization test: Compare the data with labels and the data with replacement labels on the same trained model. If there is no difference in the saliency map, it means that the saliency map method does not depend on the relationship between the image and the label.

Another of their finding is that the image processing algorithm edge detection can also have a visual effect similar to saliency map, because it can extract the edge where the gradient is significant. Meanwhile, it does not rely on deep learning models or training data. This comparison indicates that visual analysis is not so sufficient and effective in judging whether an interpretation method is sensitive to models or data. The quantitative methods should be applied in the evaluation.

Therefore, the intention of so-called sanity check is to remove some explanation methods that are not sensitive to changes in models and data before implementing a specific method.

Area over Perturbation Curve

Explanation methods (also called analyzers, methods that analyzes the model) perform differently depending on the model, the task at hand, the data, etc. In order to quantitatively evaluate those analyzers, [Samek et al., 2016] build upon the perturbation analysis, originally designed to assess explanation methods in classification networks. Let us first describe the perturbation process and then the evaluation metric.

First, the input image to be analyzed is subsampled by a grid. Each subwindow of the grid is ranked according to its importance w.r.t. to the pixel-wise saliency scores assigned by the analyzers. Then, the information content of the image is gradually corrupted by adding perturbation (Gaussian noise) to each subwindow, starting

with the most relevant subwindow, w.r.t. the ranking just mentioned. The effect of this perturbation on the model performance is measured with the prediction error. This procedure is repeated for each subwindow.
Generally, the accuracy of model will drop quickly when important information is removed and remains largely unaffected when perturbing unimportant regions. Thus, the analyzers can be compared by measuring how quickly their performance drops. That is to say, the quicker the model performance drops after introducing perturbation, the better the analyzer is capable of identifying the input components responsible for the output of the model.
The quantitative evaluation proposed in [Samek et al., 2016] for classification network, consists in computing the difference between the score $f(x)$ indicating the certainty of the presence of an object in the image x, in the presence and in the absence of perturbation. This difference is called Area over Perturbation Curve (AOPC) and defined more precisely defined in in [Samek et al., 2016] as:

$$\mathrm{AOPC}_{\mathrm{A}nalyzer} = \frac{1}{\mathrm{N}} \sum_{n=0}^{\mathrm{N}} (f(x_n)^{(0)} - \frac{1}{\mathrm{K}} \sum_{k=0}^{\mathrm{K}} f(x_n)^{(k)}) \tag{2.35}$$

where N is the number of images, K is the number of perturbation steps, x is the input image.

Input variant

[Kindermans et al., 2019] checks the reliability of saliency methods by preprocessing the input images, and they found that saliency methods that do not satisfy input invariance so that result in misleading attribution. This indicates that the saliency methods sometimes are not one hundred percent reliable so that inspire people to either abandon this method or seeks for other methods instead of saliency methods.

2.3.5 Perspectives

Besides saliency methods, graph knowledge can be used in machine learning which can augment (intermediate) features with more semantics [Lecue, 2020]. The mentioned above explanation methods are mainly post-hoc methods, that is to say the deep learning methods themselves are not explainable. There is one voice claiming that the deep learning models should be self-explainable [Rudin, 2019].

Table 2.4 – Summary of explanation methods and their abbreviations.

Method	Abbreviation	Method	Abbreviation
Class Activation maps[Zhou et al., 2016]	CAM	Anchors[Ribeiro et al., 2018]	Anchors
Prediction Difference Analysis[Zintgraf et al., 2017]	PDA	Contextual Prediction Difference Analysis[Gu and Tresp, 2019]	CPDA
Shapley Value Sampling[Castro et al., 2009]	SHAP	DeConvNet[Zeiler and Fergus, 2014]	DeConvNet
Excitation Backprop[Zhang et al., 2018]	EBP	Guided backpropagation[Springenberg et al., 2015]	GBP
NeuronGuidedBackprop[Springenberg et al., 2015]	NGBP	ExtremalPerturbation[Fong et al., 2019]	EP
MeaningfulPerturbation[Fong and Vedaldi, 2017]	MP	Occlusion[Zeiler and Fergus, 2014]	OCC
Influence[Leino et al., 2018]	Influence	Representation Erasure[Li et al., 2016]	RE
Gradient[Simonyan et al., 2013]	Gradient	Gradient*Input[Shrikumar et al., 2016]	GI
FullGrad[Lundberg and Lee, 2017]	FG	Grad-CAM[Selvaraju et al., 2017]	GC
Grad-CAM++[Chattopadhay et al., 2018]	GC++		
Guided Grad-CAM[Selvaraju et al., 2017]	GGC	SmoothGrad[Smilkov et al., 2017]	SG
VarGrad[Adebayo et al., 2018a]	VG	Intergrated Gradients[Sundararajan et al., 2017]	IG
GradientSHAP[Lundberg and Lee, 2017]	GS	Expressive Gradients[Yang et al., 2019]	EG
Deep SHAP[Chen et al., 2021a]	DS	SHAP Interaction Index[Lundberg et al., 2020]	SII
Kernel SHAP[Lundberg and Lee, 2017]	KS	DeepLIFT SHAP[Lundberg and Lee, 2017]	DLS
DeepLIFT[Shrikumar et al., 2017]	DeepLIFT	TreesExplainer[Lundberg et al., 2020]	TE
GNNExplainer[Ying et al., 2019]	GNNE	GNNLRP[Schnake et al., 2020]	GNNLRP
Layer-wise Relevance Propagation[Bach et al., 2015]	LRP	Spectral Relevance Analysis[Anders et al., 2019]	SRA
DeepTaylor[Montavon et al., 2017]	DT	LayerConductance[Shrikumar et al., 2018]	LC
Local Rule-based Explanations[Guidotti et al., 2018]	LRE	LIME[Ribeiro et al., 2016]	LIME
STREAM[Elenberg et al., 2017]	STREAM	RISE[Petsiuk et al., 2018]	RISE
PatterenNet[Kindermans et al., 2018]	PTN	Pattern Attribution[Kindermans et al., 2018]	PTNA
FIDO[Chang et al., 2019]	FIDO	NeuronConductance[Dhamdhere et al., 2019]	NC
TotalConductance[Dhamdhere et al., 2019]	TC	DeepDreams[Mordvintsev et al., 2015]	DD
TCAV[Kim et al., 2018]	TCAV	TCAV with RCV[Graziani et al., 2018]	TCAVR
UBS[Yeche et al., 2019]	UBS	SENN[Adebayo et al., 2018a]	SENN

Part II

Contributions

Chapter 3

Kappa loss for skin lesion segmentation

Contents

3.1 Motivation

Today, CNN are the state-of-the-art in medical image segmentation. One key component of CNN is the loss function, that drives the backpropagation of the error between the predicted value and and the reference label. Cross Entropy is a widely used, standard loss function. However, as mentioned in previous chapter (Chapter 2.1), class imbalance problem is prone to happen especially in skin images, in which the lesion only takes a small proportion in the whole image. In order to handle class imbalance, weights can be assigned to samples of different classes. The Dice loss function [Milletari et al., 2016], a soft approximation of the well-known Dice metric, is specifically designed for image segmentation. However, the Dice loss only considers foreground (i.e. object) pixels, and does not take into account the background pixels in the image.

Therefore, in this work, we propose a loss function called Kappa loss, based on the Kappa index, that can not only deal with class imbalance problems in medical image segmentation, but also considers the whole information of an image. The motivating factor spurring our approach is rooted in the fact that all pixels should be taken into account, since a large part of the image is occupied by object (or in our case melanoma) pixels. We believe that by using the Kappa loss, we will enforce the constrain on the true negative pixels in addition to the true positive ones, just reaching a better balance between the two classes. We demonstrate the efficiency of the proposed loss on images of skin lesions or moles, which are typical cases where the lesion takes a significant part of the image.

3.2 The clinical problem: skin cancer detection from lesion segmentation

3.2.1 Diagnosis of skin lesion

Melanoma is a type of skin cancer, that, if not detected and treated within limited time, may be fatal, since it can spread to other organs quickly. According to a survey [Allan, 2019], in 2019, about 7230 people (4740 men and 2490 women) will lose their lives because of melanoma in the USA. Dermatologists establish their diagnosis by visual inspections of moles, and by extracting texture, size and shape analysis infor-

mation. One first step is often the segmentation of the mole. Manual segmentation by dermatologists as described in [Jafari et al., 2016] is a time-consuming process, hardly compatible with the usual workload of medical experts, and that can be subjective.

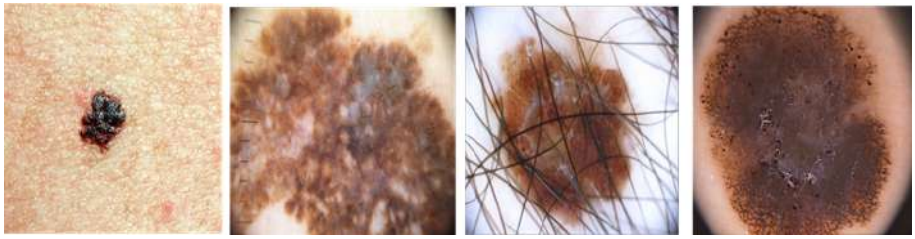


Figure 3.1 – Samples of skin images: 1st column is from dataset Skin-Cancer-Detection, 2nd column from ISIC 2017, 3rd&4th column from ISIC 2018. In the skin image of 2-4th column, the object is large with respect to the image.

3.2.2 Related works in skin lesion segmentation

For the skin lesion image segmentation, early works utilized computer vision based methods. Computer vision based image segmentation methods have been thoroughly investigated before the advent of deep learning. [Yuan et al., 2009] uses active contour to detect border of skin lesion. In [Zhou et al., 2011], the authors combine gradient vector flow with mean shift to segment the skin lesion images. Preprocessing is performed in [Schaefer et al., 2011] to tackle the problem of low contrast and color between background and object. [Pennisi et al., 2016] presents Delaunay Triangulation to extract the contour of skin lesion image. Region merging based approach is used in [Wong et al., 2011]. In [Jain et al., 2015], authors apply image processing tools to extract features (Asymmetry, Border, Color, Diameter) of skin lesion in order to classify the image as melanoma or not. Later on, deep learning based skin image segmentation methods were proposed. The work of [Jafari et al., 2016] includes 3 steps: preprocessing (image filtering), CNN, selection of largest area. In [Attia et al., 2017], the authors spend effort on aggregating convolutional and recurrent neural networks. In [Yuan et al., 2017], authors implement a CNN with a loss function based on Jaccard distance that can deal with class imbalance problem. [Xie et al., 2020a] design a deep learning architecture using attention mechanism that can generate high-resolution feature maps to preserve spatial details.

3.3 The Kappa loss

3.3.1 From metrics to loss

The Dice index (DI) is widely known as an overlap measure in binary image segmentation, defined as the ratio between twice the intersection of two regions over their union. The DI values range from 0 to 1, 0 meaning empty overlap while 1 indicates perfect match. The Dice Index was originally designed to be an inter-rater agreement [Dice, 1945], independently from the pixel labeling problem.

Table 3.1 – Counts of agreement and disagreement from two raters. $a+d$ is the number of targets for which two raters agreed, $b+c$ is the number of targets for which they disagreed, N = $a+b+c+d$.

		Ground Truth (Rater 1)		
		+	-	Total
Predicted	+	a	b	$a+b$
Results	-	c	d	$c+d$
(Rater 2)	Total	$a+c$	$b+d$	N

Let us define as a the number of counts where raters agree positively, d the number of counts where raters agree negatively, and b and c where the two raters disagree with each other, N is the sum of a, b, c and d (see Table 3.1).
In the image segmentation, we regard the two raters as ground truth and the predicted image, Figure 3.2 shows the interpretation of number of counts with respect to the segmentation problem. More specifically, the element a is foreground (represented by positive "+") shared by both of ground truth and predicted area, b is foreground of predicted image and background of ground truth, c is foreground of ground truth and background of predicted image, d is background (represented by negative "-") shared by both of ground truth and predicted area.
According to the definition of Dice index and the distribution of Venn diagram (Figure 3.2), one can write the DI as:

$$\mathrm{DI} = \frac{2a}{2a+b+c} \tag{3.1}$$

Note that in Equation 3.1, the d counts where raters disagree negatively are not taken into account in this definition. While other agreement rates can take into account these true negative. In particular, the *Kappa* coefficient [Hubert, 1977], a

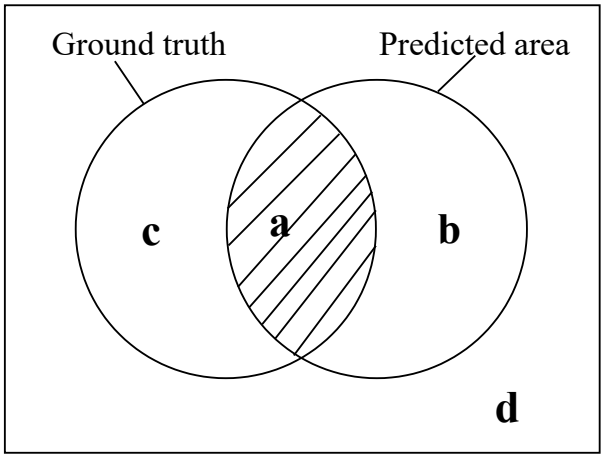


Figure 3.2 – Venn diagram of ground truth and predicted area. Ground truth contour vs predicted contour, with a, b, c the number of pixels included in both contours, only in the predicted area, only in the ground truth, respectively.

chance-corrected measure of agreement voted by two raters, is defined as:

$$\mathrm{K}appa = \frac{2(ad - bc)}{(a+b)(b+d) + (a+c)(c+d)} \tag{3.2}$$

As recalled in the pioneering paper [Zijdenbos et al., 1994] that first uses the Dice index as a metric to evaluate segmentation quality, the Dice index is a limit case of the *Kappa* index when $d \gg a, b, c$:

$$\lim_{d \to \infty} \mathrm{K}appa = \frac{2a}{2a+b+c} = \mathrm{DI}, \tag{3.3}$$

Thus, that is to say, the Dice index only considers the foreground pixels to compute the overlap of the predicted region and the ground truth, based on the assumption that region or object pixels are small compared to the background area. However in some cases, especially in medical skin images, this assumption does not hold. We show examples of such cases, in the 3 images on the right in Figure 3.1. This is the rationale behind the use of the kappa index as loss function: all pixels in the image are taken into account, and not only the foreground pixels. Note that a weighted version of the kappa index has shown to be a loss of choice for ordinal classification, in comparison to logarithmic loss [de La Torre et al., 2018]. However, it was not introduced in the context of image segmentation.

3.3.2 Definition of the Kappa loss

In order for the Kappa coefficient to be used as a loss function in a CNN, it has to be differentiable so that its gradient may be computed. Thus the probabilities (i.e. output values of the last layer of the networks) have to be used, instead of hard labels, in the definition of the Kappa loss. We rewrite elements a, b, c and d by taking into account the predicted segmentation (or probability) at pixel i, denoted as p_i, and the ground truth at this same pixel, denoted as g_i. The pixel-wise representation of these elements is shown in Formula 3.4, where N is the number of pixels in the image.

$$\begin{aligned} a &= \sum_{i=1}^{\mathrm{N}} (p_i g_i), \\ b &= \sum_{i=1}^{\mathrm{N}} (1-p_i) g_i, \\ c &= \sum_{i=1}^{\mathrm{N}} (1-g_i) p_i, \\ d &= \sum_{i=1}^{\mathrm{N}} (1-p_i)(1-g_i) \end{aligned} \tag{3.4}$$

We obtain the soft approximation of the Kappa loss by replacing the affectations from Formula (3.4) in Formula (3.2).

$$\mathrm{K}appa\ \ loss = 1 - \frac{2(ad-bc)}{(a+b)(b+d)+(a+c)(c+d)} \tag{3.5}$$

Then, substituting the soft proxy of Equation 3.4 into Kappa loss (Formula 3.5) and simplifying it, we get:

$$\mathrm{K}appa\ \ loss = 1 - \frac{2\sum_{i=1}^{\mathrm{N}} p_i g_i - \sum_{i=1}^{\mathrm{N}} p_i \cdot \sum_{i=1}^{\mathrm{N}} g_i/\mathrm{N}}{\sum_{i=1}^{\mathrm{N}} p_i + \sum_{i=1}^{\mathrm{N}} g_i - 2\sum_{i=1}^{\mathrm{N}} p_i g_i/\mathrm{N}} \tag{3.6}$$

Deriving the Kappa loss with respect to predicted probabilities at pixel j, the gradient of Kappa loss is as Formula 3.7:

$$\begin{aligned} \frac{\partial \mathrm{K}appa}{\partial p_j} = &-2\frac{g_j(\sum g_i + \sum p_i - 2\sum p_i \sum g_i/\mathrm{N})}{(\sum p_i + \sum g_i - 2\sum p_i g_i/\mathrm{N})^2} \\ &+ \frac{\sum p_i g_i (1 - 2\sum g_i/\mathrm{N}^2) + (\sum g_i)^2/\mathrm{N})}{(\sum p_i + \sum g_i - 2\sum p_i g_i/\mathrm{N})^2} \end{aligned} \tag{3.7}$$

In the case of Dice loss, Formula (3.6) of Kappa loss boils down to Dice loss [Milletari et al., 2016]:

$$\text{Dice loss} = 1 - \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} g_i} \tag{3.8}$$

For this Dice loss, variants of this definition may have $p_i^2 + g_i^2$ instead of $p_i + g_i$ in the denominator, or include a smoothness term (a small value) added to the denominator and the numerator [Sudre et al., 2017, Wong et al., 2018, Pedemonte et al., 2018], but that only helps in case of missing labels and is not critical. At this point, we have verified the theoretical derivation of Kappa loss and its derivability. Next, we will verify its feasibility and performance in combination with U-Net in our experiments.

3.3.3 CNN for image segmentation

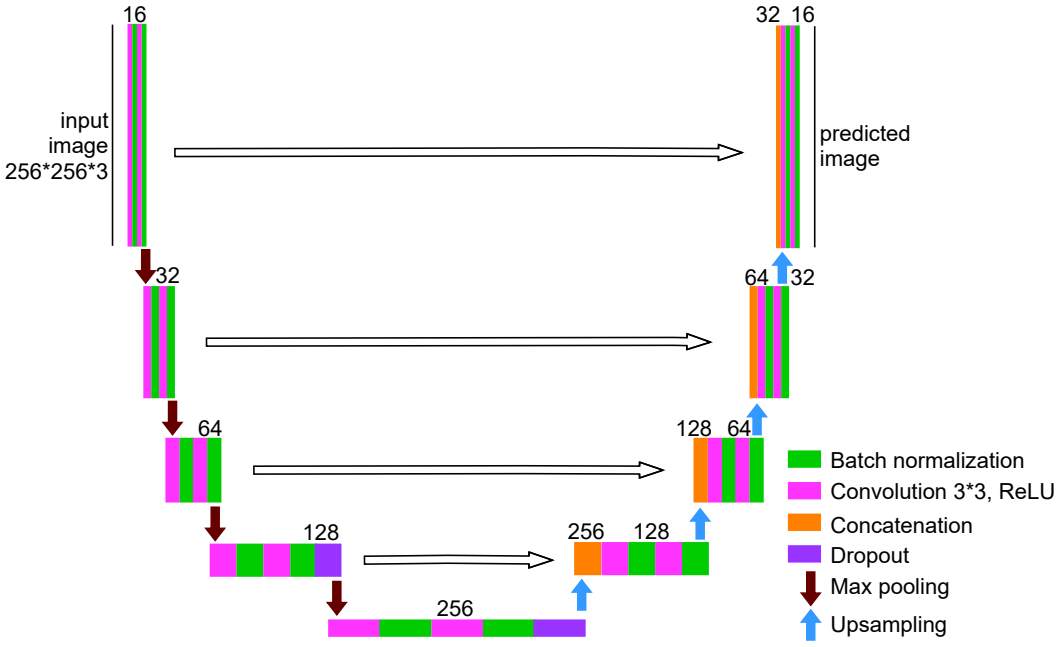


Figure 3.3 – Architecture of customized U-Net. Each box corresponds to a multi-channel feature map. The number of channels is denoted above the box, different colors mean different operation.

We have used the well-known U-Net architecture, one of the most popular CNN for medical image segmentation [Ronneberger et al., 2015], to implement the Kappa

loss function. The U-Net is a fully convolutional network with an encoder-decoder architecture and skip connections. Compared to the original architecture, we simplified the network, given the limited amount of images. The original U-Net includes 64 filters at the first level, for a total of 31,031,685 parameters. We set the initial number of filters to be 16 (3×3), so the number of parameters is 1,946,449. We also add a batch normalization [Ioffe and Szegedy, 2015] operation after each ReLU activation function to avoid gradient vanishing. In order to avoid overfitting, we use drop out to reduce parameters in the U-Net with a rate of 0.5.

3.4 Experiments and results

3.4.1 Datasets

We use 6 publicly available datasets of skin images with mole or melanoma, to assess the proposed Kappa-based loss function. They are: Skin-Cancer-Detection (SCD, 206 images, supplied by Vision and Image Processing Lab, University of Waterloo), split into two subsets which are melanoma (Mel, 119 images) and not-melanoma (Non-mel, 87 images), and 3 datasets from International Skin Imaging Collaboration (ISIC) [Codella et al., 2018, Tschandl et al., 2018, Codella et al., 2017, Gutman et al., 2016] which have 2594, 2000 and 900 images respectively. In the latter, images not only include a lesion part but also present noise such as hair, which increases the difficulty of segmentation. Moreover, the object (lesion area) in the image of dataset SCD or ISIC 2018 varies from one to another in size. We split each dataset into training set, validation set and test set, respectively, with the same amount of images in each set. Images are resized to 256 × 256.

3.4.2 Experimental settings

Because the amount of medical data is limited, we use data augmentation to increase the number of it. Data augmentation including rotation, shifting, shearing, zooming and flipping is used in training. Protocol is a 3-fold cross validation. The optimizer is Adam [Kingma and Ba, 2014] with a learning rate of $1e^{-4}$. The batch size is 4. The model is trained for 100 epochs. Evaluation metrics are the Dice index (DI) and the Hausdorff distance (HD), which is the maximum point-to-point distance between two contours. The implementation tool is based on Keras and Tensorflow 1.0. Tesla P100 GPU server is used in the experiments.

3.4.3 Results

Quantitative analysis

We trained the U-Net described in the previous section from scratch on the 6 datasets independently, with two different loss functions, the Dice loss that will be the baseline, and the Kappa loss. Results are shown in Table 3.2. The Hausdorff distances between Kappa and Dice losses are similar, which means that Kappa is not correcting distant, false positive pixels, except for the first dataset (Non-mel), where HD drops by 7%. However, substantial improvement on the Dice Index (DI) is obtained for the Kappa loss, in comparison to the Dice loss, for several datasets.

Table 3.2 – Averaged Dice index (DI) and Hausdorff distance (HD) values (± standard deviation), for Dice and Kappa losses on 6 different datasets (87, 119, 206, 900, 2000, 2594 images respectively).

	Dice loss		**Kappa loss**	
dataset	**DI ↑**	**HD(mm) ↓**	**DI ↑**	**HD(mm) ↓**
Non-mel	0.65±0.11	5.06±1.79	0.73±0.11	4.70±2.02
Mel	0.80±0.06	6.70±1.93	0.81±0.03	6.59±1.88
SCD	0.82±0.04	7.94±1.72	0.83±0.03	7.91±1.68
ISIC-16	0.80±0.05	8.42±2.19	0.84±0.01	8.41±2.25
ISIC-17	0.80±0.05	8.07±1.93	0.84±0.05	8.03±1.94
ISIC-18	0.81±0.03	7.59±2.60	0.82±0.04	7.52±2.66

Qualitative analysis

Some segmentation results are shown in Figure 3.4, which shows that in some cases, the Kappa loss can help to make the segmentation more accurate. Looking at Figure 3.5, we can also observe that the Kappa loss converges faster than Dice loss under the same U-Net model settings.

Key feature maps of Kappa loss

According to example images in Figure 3.6, we extract several feature maps of U-Net from different layers with Dice loss and Kappa loss respectively (Figure 3.7-3.12). Such that we can know the intermediate inference process. These three skin lesion images are with noise, small lesion and large lesion area from dataset ISIC 2016. Those feature maps are selected from encoder and decoder of U-Net. One can find

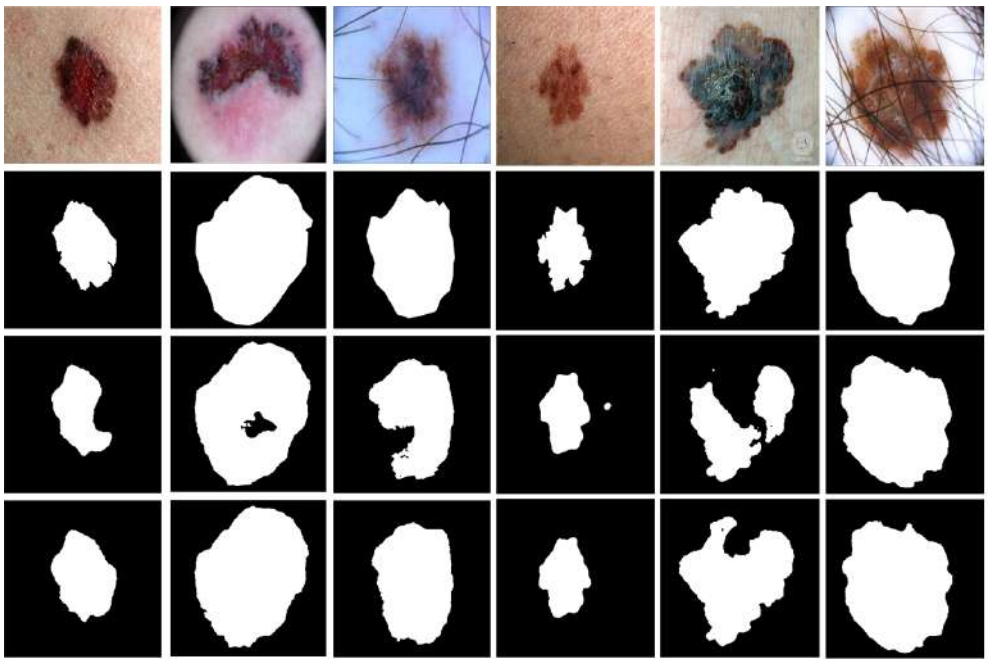


Figure 3.4 – Examples of segmentation results. From up to down: skin lesion image, ground truth, segmentation result with Dice and Kappa loss.

that the inner world of model with Dice loss and Kappa loss have different attentions. The U-Net with Kappa loss seems focus more on skin lesion itself. While the U-Net with Dice loss is a little more distracting (Features around the lesion area are also emphasized.) This may be due to the fact that Kappa loss has background pixels as a constraint term compared to Dice loss, thus making the Kappa loss function more stringent for each weight of neuron update during the backpropagation.

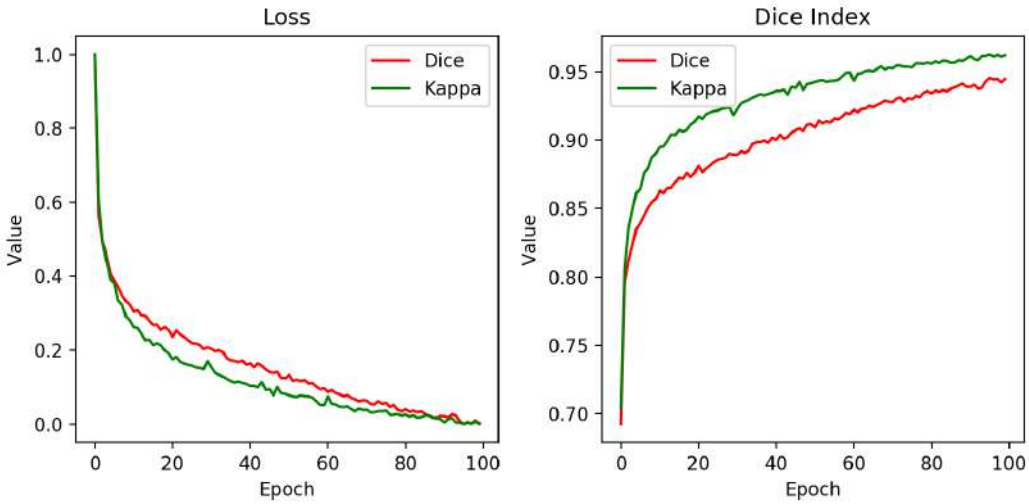


Figure 3.5 – Loss function (left) and DI metric (right) during the 100 epochs of training process on the dataset ISIC 2016.

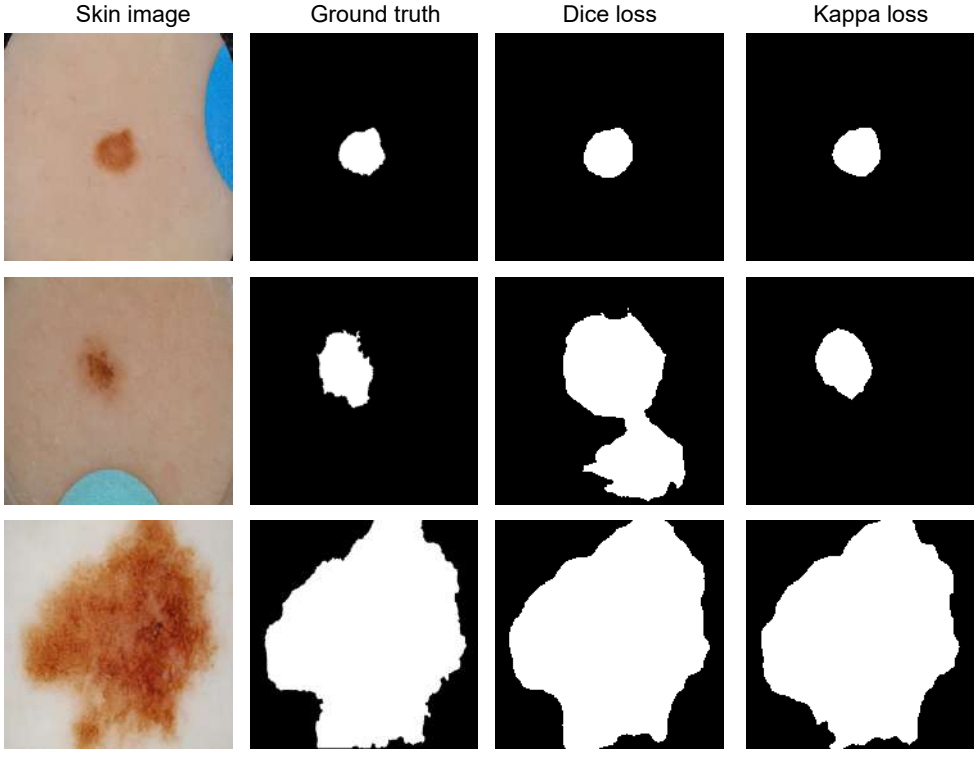


Figure 3.6 – Skin lesion segmentation results of U-Net with Dice loss and Kappa loss respectively.

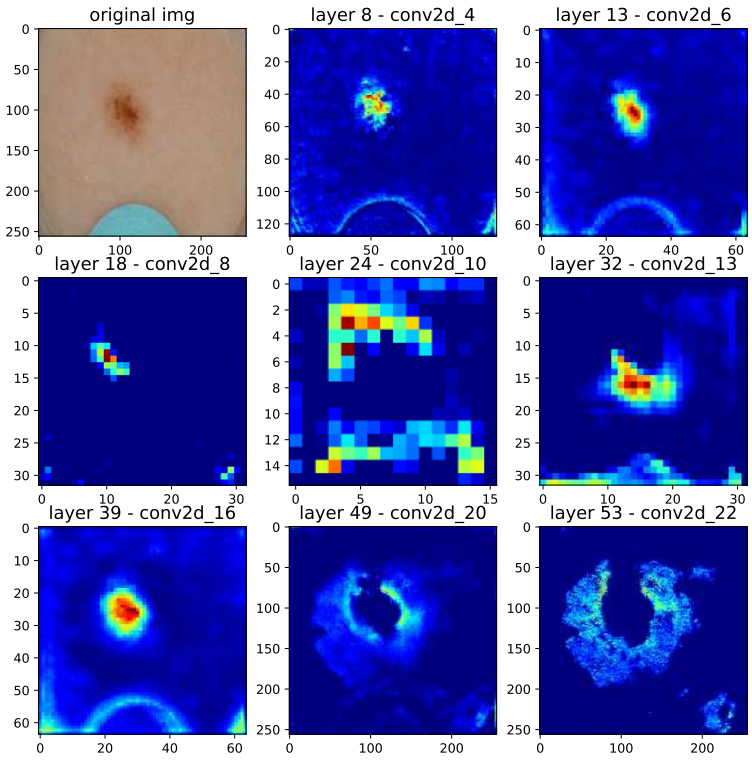


Figure 3.7 – Feature maps of U-Net with Dice loss on a noisy skin lesion image.

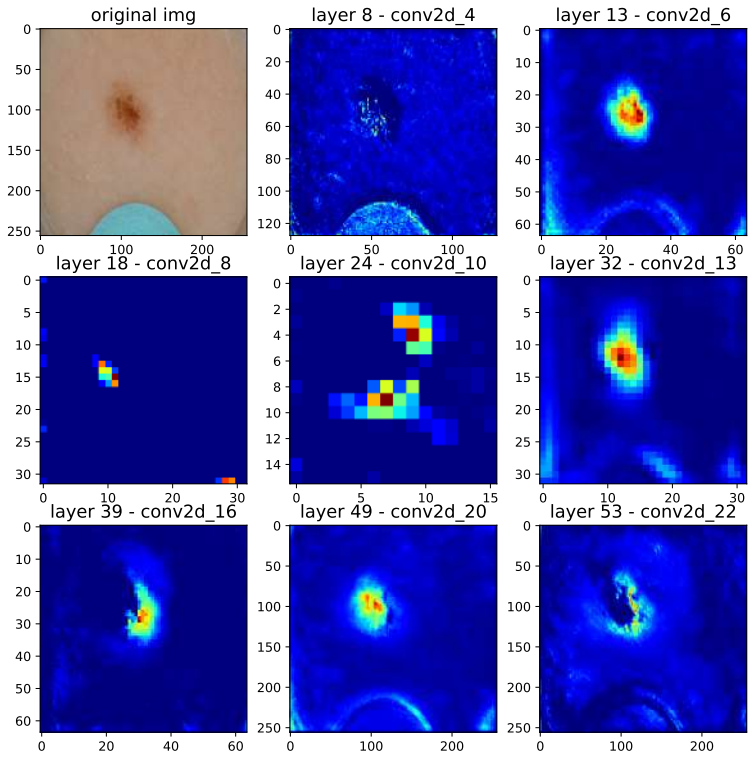


Figure 3.8 – Feature maps of U-Net with Kappa loss on a noisy skin lesion image.

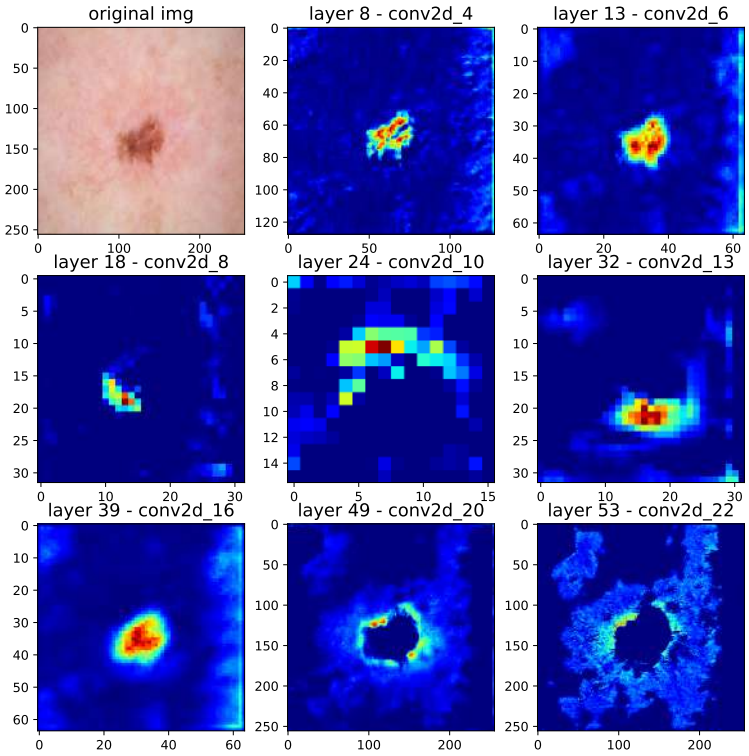


Figure 3.9 – Feature maps of U-Net with Dice loss on a skin image with small lesion.

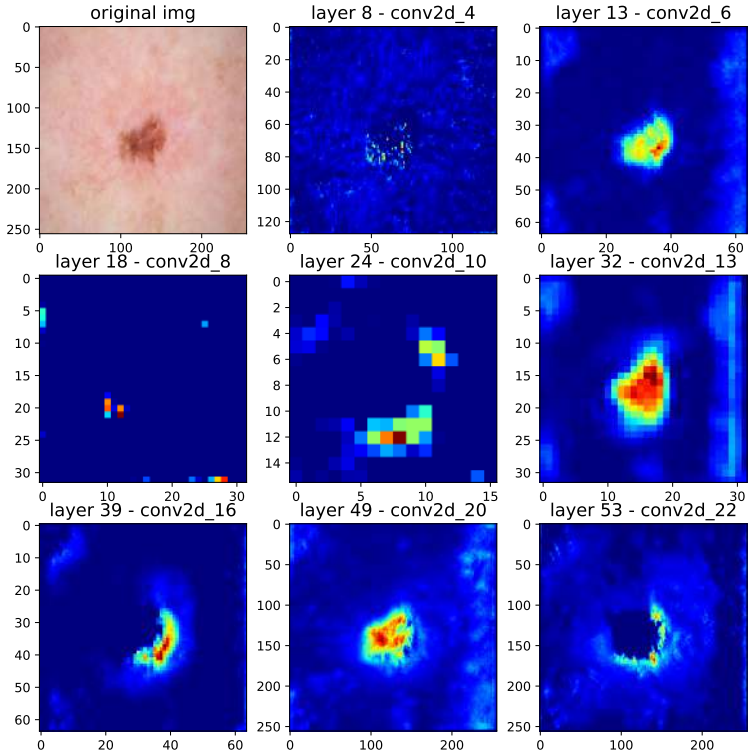


Figure 3.10 – Feature maps of U-Net with Kappa loss on a skin image with small lesion.

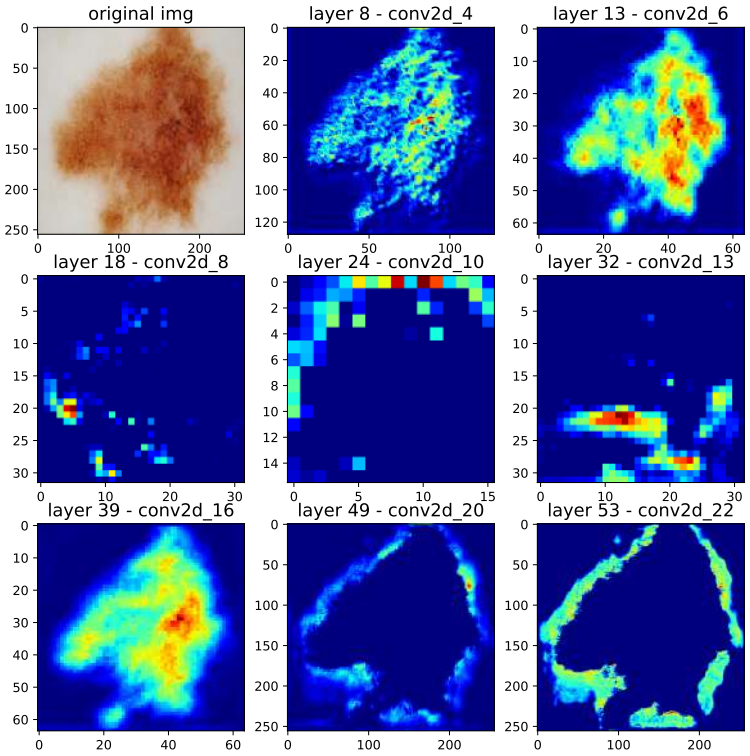


Figure 3.11 – Feature maps of U-Net with Dice loss on a skin image with large lesion.

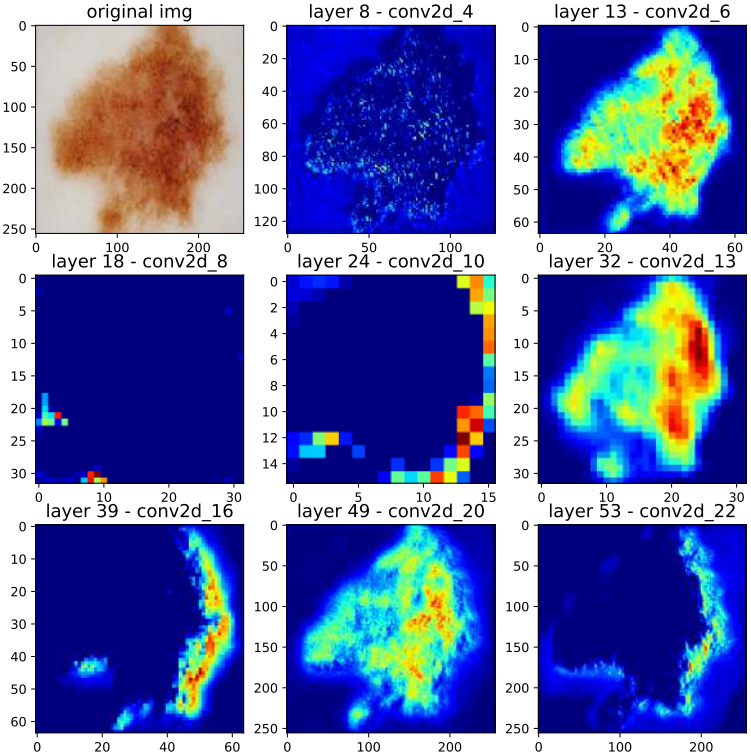


Figure 3.12 – Feature maps of U-Net with Kappa loss on a skin image with large lesion.

3.5 Conclusion

In this work, we have proposed a new loss function, based on the Kappa index, to be used in CNN for medical image segmentation. Different from the Dice loss, this loss function considers all pixels (background pixels included) in the evaluation of the predicted segmentation. We believe that by enforcing constraint on both positive and negative pixels, segmentation accuracy or convergence may be improved. We have shown the Kappa loss differentiability and used the state-of-the-art U-Net architecture to implement it. We compared the Kappa loss quantitatively to the Dice loss on several public datasets of melanoma and skin segmentation. Promising results were obtained, showing the potential of the Kappa loss. Future work involves extending our benchmarking experiments to other loss functions, to further investigate the behavior of Kappa loss with respect to other loss functions; and generalizing the Kappa loss to multi-label image segmentation, as was proposed for the generalized Dice loss in [Sudre et al., 2017]. Kappa loss could also be used in multi-scale approaches, when segmentation is required inside a region of interest (e.g. bounding box), where there is a balance between positive and negative pixels.

Chapter 4

Fetus head circumference prediction

Contents

4.1 Motivation

Very often, medical image segmentation is the first step to compute parameters from the image, such as volume: for example, the cardiac ventricles are segmented in magnetic resonance images in order to estimate the cardiac contractile function via some indices (e.g. ejection fraction) [Petitjean and Dacher, 2011]. Another example is anthropometry, where measuring the skeletal muscle body mass and fat body mass, which is a significant pronostic factors in cancer, are estimated from the segmentation of muscle and fat in CT images [Zhen et al., 2015b, Hussain et al., 2016, Pang et al., 2018, Luo et al., 2020a, Zhang et al., 2020a, Zhang et al., 2020b].

Instead of resorting to segmentation, which is a costly and error-prone process, one can attempt to estimate the (single or multiple) characteristics or biomarkers, directly. Works on this topic have gotten a second wind with the breakthrough of deep learning, that allows to take advantage of the power of feature representation and to perform an end-to-end regression. However direct, "segmentation-free" approaches rely on much less information to estimate the biomarker, and it is not clear yet if segmentation-free approaches can reach the level of accuracy of segmentation-based approaches. To our knowledge, there is no study that rigorously compares segmentation based methods and segmentation-free methods for a given application of biomarker estimation, and quantifies the gap between them. This observation motivates the present contribution, where we propose a fair, quantitative comparison of segmentation-based and segmentation-free (i.e. regression) approaches to estimate how far regression-based approaches stand from segmentation approaches, for a problem that has a major clinical impact: the estimation of the head circumference in US images. This estimation is important to accurately assess the growth of the fetus.

In this chapter, we investigate several settings, i.e. state of the art segmentation models and various backbones for the regression CNN architectures, to obtain the best of both worlds, and investigate also time and memory consumption in addition to estimation accuracy. To make the segmentation-free approaches more convincible, we adapt explanation methods in regression CNN and provide an interpretation of what a saliency map is, in the regression case. We are thus able to gain insight into the CNN regression model for our HC prediction problem, and see what pixels contribute the most to the estimation of the HC: we expect them to be those of the head

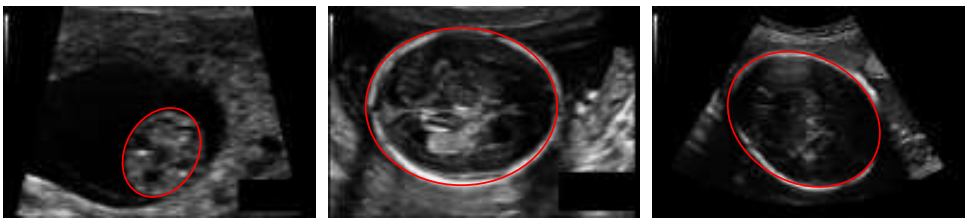
contour. We also address the problem of evaluating the explanation methods, in the regression case. Adebayo's sanity checks consist in performing randomization tests, in the data or in the model, and evaluate the changes in the produced saliency maps [Adebayo et al., 2018b]. Another example is Samek's proposal, that has particularly inspired us [Samek et al., 2016], to compare and assess different explanation methods. The principle is to inject noise gradually in the image, in locations that have been highlighted by the saliency maps, and see how the prediction is affected by this perturbation. However, the method is designed for classification networks and requires some adaptation.

4.2 HC measurement from US images

4.2.1 Background

Automated measurement of fetus head circumference (HC) is performed throughout the pregnancy as a key biometric to monitor fetus growth and estimate gestational age. In clinical routine, this measurement is performed on ultrasound (US) images, via manually tracing of the skull contour, along to fitting it to an ellipse, this being done by sonographers. Figure 4.1 is one sample of fetus head of ultrasound (US) image, from the HC18 public dataset [van den Heuvel et al., 2018b] used in this paper. Identifying the head contour is challenging due to low signal-to-noise ratio in US images, and also because the contours have fuzzy (and sometimes missing) borders (Figure 4.1). Manual contouring is an operator-dependant operation, which is measured by experienced sonographers by calipers, subject to intra and inter-variability, which yields inaccurate measurements, as measured in [Sarris et al., 2012]: the 95% limits of agreement have been measured to ±7mm for the intra-operator variability and ±12mm for the inter-operator variability. Another study concluded that the sonographic measured HC consistently underestimates the actual postpartum HC by an average of 13.5 mm or 4% [Melamed et al., 2011].

Usually, automating the measurement of fetus head circumference in US images is achieved through automatic segmentation methodology. Segmentation methods typically involves image-processing or machine learning based approaches, some post-processing of the result, so as to fit it into an ellipse. This process involves multiple steps, is adhoc, and can be prone to error. Let us emphasize on the fact that here, the segmentation is just an intermediate step to compute a characteristic



44.3 mm
640.7 pixels

160.7 mm
1404.8 pixels

272.9 mm
1157.0 pixels

Figure 4.1 – US images of fetus head from HC18 dataset [van den Heuvel et al., 2018b]. Red ellipses are head contours. Below the image is given the corresponding head circumference (HC). Images may have different pixel size.

from the image, i.e. the length of the head contour.

4.2.2 Related Works on automated head circumference estimation from US images

Several approaches have been proposed in the literature to measure the head circumference in US images, based on image segmentation [Li et al., 2017, Lu et al., 2005, Jardim and Figueiredo, 2005]. Some follow a two-step approach, namely fetus head localization and segmentation refinement. For example, in [van den Heuvel et al., 2018a], the first step consists in locating the fetus head with Haar-like features used to train a random forest classifier; and the second step consists in the measurement of the HC, via ellipse fitting and Hough transform. Similar method is used in [Li et al., 2017].The above segmentation method is based on traditional machine learning. In recent years, deep learning-based head circumference segmentation algorithms have improved in terms of performance and efficiency. These approaches build upon deep segmentation models also in a two-step process, contour prediction and ellipse fitting [Kim et al., 2019]. In [Budd et al., 2019], the standard segmentation model U-Net [Ronneberger et al., 2015] is trained using manually labeled images, and segmentation results are fitted to ellipses. The mean absolute error (MAE) tested on HC18 dataset in [Budd et al., 2019] is 1.90 mm, the Dice accuracy is 0.982, the Hausdorff distance (HD) is 1.292 mm. In [Sobhaninia et al., 2019], authors build upon the same idea, combining image segmentation and ellipse tuning together in a multi-task learning network. Their segmentation accuracy is 0.968 in Dice score, 1.72 mm in HD, 2.12 mm in MAE. In [Fiorentino et al., 2021], the authors use first a region-proposal CNN for head lo-

calization, and a regression CNN trained on distance fields to segment the HC. [Moccia et al., 2021] advances the work [Fiorentino et al., 2021] since they propose Mask-R^{2}CNN neural network to perform HC distance-field regression for head delineation in an end-to-end way, which does not need prior HC localization or post-processing for outlier removal. All these methods rely on a segmentation of the fetus head as a prerequisite to estimating the HC.

The segmentation free approaches for biomarker estimation have been introduced in Chapter 2. Throughout the above literature, there are no studies based on ultrasound images and there are no methods to directly measure fetus head circumference. Therefore, based on the above studies that have been successfully implemented for objects such as the heart, one of the core tasks of this thesis is to find a scheme to directly predict fetus head circumference, to explore the interpretability of the method, and to compare it with segmentation-based methods.

4.3 Methodological framework

Figure 4.2 shows both architectures of segmentation-based and segmentation-free (regression-based) approaches. We will describe these two models in detail in the following sections.

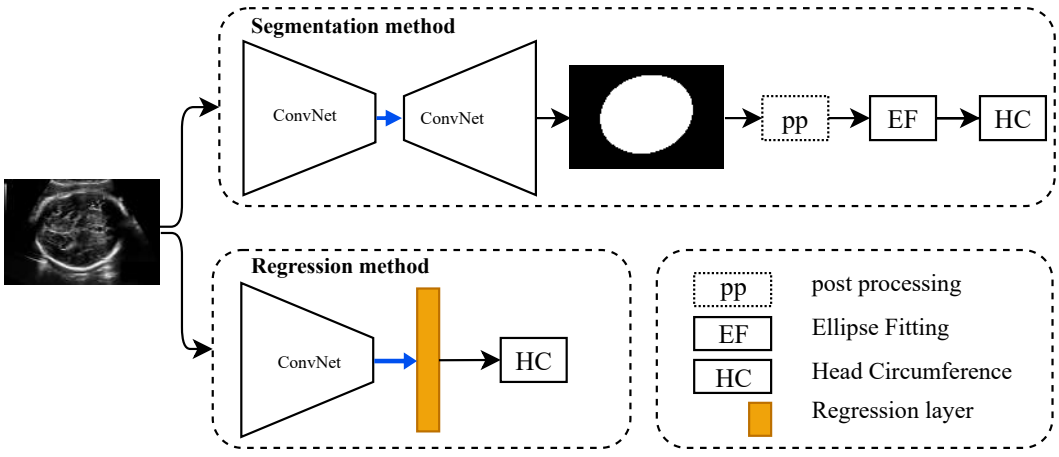


Figure 4.2 – Overview of head circumference estimation process based on either deep segmentation-based method or deep regression-based method. HC: Head circumference, pp: post-processing (The dotted box means it is optioanal), EF: Ellipse fitting.

4.3.1 Head circumference estimation based on segmentation

CNN segmentation model

We investigate several segmentation architectures which are the state of the art network in medical image segmentation, to segment the contour of fetus head: the well-known U-Net model [Ronneberger et al., 2015], U-Net++ [Zhou et al., 2018], DoubleU-Net [Jha et al., 2020], FPN [Lin et al., 2017a], LinkNet [Chaurasia and Culurciello, 2017], PSPNet [Zhao et al., 2017]. We trained these architectures from scratch but also investigate transfer learning as a way to mitigate the limited number of images in the HC18 dataset. Even though the natural images from ImageNet[1] [Deng et al., 2009] and US images have obvious dissimilarities, some generic representations can be learnt from a large-scale dataset, that might be beneficial to other types of images, and they have proven so in the context of MR images [Wacker et al., 2020]. Thus we have used various backbone models, namely VGG16 [Simonyan and Zisserman, 2015], ResNet50 [He et al., 2016a], EfficientNet [Tan and Le, 2019], pre-trained on the ImageNet dataset, for all architectures mentioned above. For the loss function, we use the Dice loss, highlighted by [Ma et al., 2021] to be one of the best loss function for medical image segmentation.

Post-processing of segmentation results

It can happen that the segmentation results have some noise or incomplete part such as holes, which can cause inaccurate ellipse fitting. Thus some post-processing is applied on the segmentation results: contours are detected from the segmentation map by Canny filter [Canny, 1986], then the largest connected component is kept when several contours are detected. Generally, the shape of the maximum contour is irregular and this randomly shaped contour needs to be fitted to an ellipse before obtaining the ellipse parameters.

HC computation based on segmentation results

To my knowledge, there are three ways to measure the length of an ellipse based on a given binary image.

[1] The most highly-used subset of ImageNet is the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012-2017 image classification and localization dataset. This dataset spans 1000 object classes and contains 1,281,167 training images, 50,000 validation images and 100,000 test images.

1. Counts the number of pixel points of the ellipse outline in the image.

2. The Euclidean distance between the locations of each contour pixel point is calculated and then accumulated to obtain the arc length of the ellipse [2].

3. Apply the formula for calculating the circumference of an ellipse.

The common drawback of the first two methods is that the calculated ellipse perimeter is larger than the actual ellipse perimeter when there are duplicate pixel points on the ellipse contour. So in this work, we use the third method of calculating the elliptical perimeter.
After post-processing the segmented results, the next step is to perform ellipse fitting in order to get the parameters (long axis, short axis, center points, angle) of the ellipse to compute its length. The length of an ellipse denoted HC is approximated by Ramanujan approximation method (Equation 4.1) [Barnard et al., 2001] in which $h = \frac{(a-b)^2}{(a+b)^2}$, a and b being the long and short axis of the ellipse:

$$\mathrm{HC} = \pi(a+b)(1 + \frac{3h}{10 + \sqrt{4-3h}}) \tag{4.1}$$

4.3.2 Head circumference estimation using regression CNN

Regression CNN model

As shown in Figure 4.2, the regression CNN are composed of a CNN backbone and regression layer (linear activation function), which can learn the features of input fetus head to estimate HC value directly. The function of CNN backbone is to extract key features from input training data. Afterwards, these feature maps are flattened into one long feature vector. The feature vector are activated by linear function (Actually keep unchanged). The backbone CNN that we experimented are state-of-the-art architectures: VGG16 [Simonyan and Zisserman, 2015] (16 in VGG16 refers to it has 16 layers that have weights.), ResNet50 [He et al., 2016a] (It is a variant of ResNet model which has 48 convolution layers along with 1 MaxPooling and 1 Average Pooling layer.), EfficientNetb2 [Tan and Le, 2019] (It has various architecture versions from b0 to b7.), DenseNet121 [Huang et al., 2017] (The number 121 corresponds to the number of layers with trainable weights (exclude batch norm)), Xception [Chollet, 2017], MobileNet [Howard et al., 2017], InceptionV3 [Szegedy et al., 2016]

[2] OpenCV library function `arcLength` is used.

(The versions including Inception V1, V2, V3, V4 and Inception-ResNet.). In order to improve model convergence, and for the reasons stated above in the previous section, these models have been pretrained on ImageNet [Deng et al., 2009], and we fine-tune[3] them for the task at hand.

Loss functions

MAE loss The loss functions commonly used in regression CNN include the Mean Absolute Error (MAE) loss (also called L1 loss), defined as:

$$\text{MAE loss} = \frac{1}{\text{N}} \sum_{i=1}^{\text{N}} |p_i - g_i| \tag{4.2}$$

Where p_i is the probability of predicted pixels, g_i the real value of head circumference in pixels, and N the number of pixels in an image. MAE loss function is more stable when dealing with outliers.
However, MAE has a serious problem (when used for neural networks): the gradient of the update is always the same, i.e., the gradient is large even for small values of loss. This is detrimental to the learning of the model. To solve this drawback, we can use a varying learning rate (e.g. Adam optimizer [Kingma and Ba, 2014]) that reduces the learning rate when the loss is close to a minimum.
MSE loss Mean Square Error (MSE) loss (L2 loss) is defined in Equation 4.3. The MSE performs well and converges effectively even with a fixed learning rate. Because the gradient of the MSE loss increases as the loss increases and decreases as the loss tends to 0. The gradient of the MSE loss increases as the loss increases and decreases as the loss tends to 0. This leads to more accurate results at the end of training using the MSE model. But, since the MSE loss takes the square of the error, so if the error > 1, then the MSE will further increase the error. If there are outliers in the data, then the error value will be large, so a model using MSE will give greater weight to the outliers compared to using MAE to calculate the loss.

$$\text{MSE loss} = \frac{1}{\text{N}} \sum_{i=1}^{\text{N}} (p_i - g_i)^2 \tag{4.3}$$

Huber loss There is another loss function called the Huber loss (HL), in which it combines the MSE loss and MAE loss through a hyper parameter δ. Here $\delta = 1$, that

[3] Fine-tuning is arguably the most widely used approach for transfer learning when working with deep learning models. Generally, it starts with a pre-trained model on the source task and trains it further on the target task [Guo et al., 2019]

is to say, when the error is less than 1, the Huber loss becomes MSE loss, otherwise, it becomes parametric MAE loss. Thus, the benefit of Huber loss can dynamically call the loss function according the change of error so that utilizing the strengths of MAE and MSE losses in suitable case.

$$\mathrm{HL} = \begin{cases} \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} \frac{1}{2}(p_i - g_i)^2, & \text{for} \quad |p_i - g_i| < \delta \\ \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} \delta * (|p_i - g_i| - \frac{\delta}{2}), & \text{otherwise} \end{cases} \tag{4.4}$$

We will investigate all three of them in our experiments, as there is no heuristic to choose one loss over the other, as highlighted in [Lathuilière et al., 2019].

Model configuration

Due to space limitations, we cannot list all the CNN models, instead we list several models with outstanding performance in recent years. In this work, several CNN backbone models in segmentation and regression respectively are utilized in order to fine tune the networks with the HC18 dataset. For segmentation models, the weights of encoder and decoder are set to be trainable, the activation function of last layer is Sigmoid function because of binary pixel-wise classification. For regression models, both the weights of CNN feature extractor part and regression layer are trainable. Because the number of training data of HC18 is limited, to avoid overfitting, we set the dropout rate as 0.7; in other words, 30% of parameters in regression CNN models are kept. The activation function of last layer in Regression CNN is linear function. The number of trainable parameters of each model is listed in Table 4.1.

4.3.3 Explainability of regression CNN

If deep learning methods are the gold standard in most image processing tasks, they are often considered as black boxes and fails to provide interpretable decisions. In this work, we investigate various saliency maps methods, to leverage their ability at explaining the predicted value of the regression CNN. Since saliency maps methods have been developed for classification CNN mostly, we provide an interpretation for regression saliency maps, as well as an adaptation of a perturbation-based quantitative evaluation of explanation methods.

Table 4.1 – Number of trainable parameters (#) of segmentation and regression CNN models. M = million. Backbone names: B1 = VGG16, B2 = ResNet50, B3 = EfficientNetb2, B4 = DenseNet121, B5 = Xception, B6 = MobileNet, B7 = InceptionV3. Reg=Regression.

Segmentation models	**# param (M)**	**Regression models**	**# param (M)**
Original U-Net	31.06	Reg-B1	15.15
U-Net-B1, B2, B3	23.75, 32.51, 14.23	Reg-B2	23.63
DoubleU-Net	29.29	Reg-B3	76.73
U-Net++ B1, B2, B3	24.15, 34.34, 16.03	Reg-B4	70.04
FPN-B1, B2, B3	17.59, 26.89, 10.77	Reg-B5	20.91
LinkNet-B1, B2, B3	20.32, 28.73, 10.15	Reg-B6	3.26
PSPNet-B1, B2, B3	21.55, 17.99, 9.41	Reg-B7	21.82

Explanation methods for CNN

In Chapter 2, we have reviewed plenty of explanation methods. In this study, we use several post-hoc explanation methods to investigate the explainability of regression CNNs. Two categories of explanation methods are generally considered, which yields a saliency map that estimates how much each pixel contributes to the prediction. They are perturbations-based or propagation-based. In perturbation-based approaches, the goal is to estimate how perturbation applied to the input image, such as blurring or injecting noise, changes the predicted class [Fong and Vedaldi, 2017, Zintgraf et al., 2017]. In propagation-based techniques, the idea is to backpropagate a relevance signal from the output to the input. In this work, we will focus on the latter category (propagation-based) of methods that actually encompass:

- (i) Sensitivity (gradient-based) methods: The **Gradient** [Simonyan et al., 2013] method; the **SmoothGrad** [Smilkov et al., 2017] method; the **Input*Gradient** [Shrikumar et al., 2016] method; and the **Integrated Gradients** [Sundararajan et al., 2017].

- (ii) Deconvolution methods. The **DeConvNet** [Zeiler and Fergus, 2014] method; the **Guided BackProp** [Springenberg et al., 2015] method.

- (iii) Layer-wise Relevance Propagation (LRP) variants: **LRP** [Bach et al., 2015] method; **DeepTaylor** [Montavon et al., 2017] method.

In the classification setting, a saliency map provides an estimation of how much each pixel contributes to the class prediction. In the regression setting, the saliency

map will provide an estimation of how much each pixel is impacting the model, and is contributing to decrease the prediction error, as measured by the loss function, that is in general the MAE or MSE.

Evaluation of explanation methods based on perturbation

In Chapter 2, we introduced some evaluation methods on various explanation methods, and Area over Perturbation Curve (AOPC) [Samek et al., 2016] is one of evaluation methods that build upon the perturbation analysis in classification tasks. Here, we propose to adapt the AOPC to the regression case. Generally, in classification or segmentation tasks, the evaluation metrics is accuracy, while in regression CNN model, the metrics is loss value between true value and predicted value. If we denote by $\epsilon(x)^{(0)}$ the prediction error of initial image evaluated by the analyzer and $\epsilon(x_n)^{(k)} (1 \leq k \leq \mathrm{K})$ the prediction error of the perturbed image $(x_n)^{(k)}$ at step k, we can define the $\mathrm{AOPC}^{\mathrm{regression}}_{\mathrm{Analyzer}}$ as:

$$\mathrm{AOPC}^{regression}_{\mathrm{A}nalyzer} = \frac{1}{\mathrm{N}} \sum_{n=0}^{\mathrm{N}} (\epsilon(x_n)^{(0)} - \frac{1}{\mathrm{K}} \sum_{k=0}^{\mathrm{K}} \epsilon(x_n)^{(k)}) \tag{4.5}$$

A larger AOPC in absolute value means that an analyzer has a steep decrease when the perturbation steps is increasing.

4.4 Experiments and results

4.4.1 HC18 Dataset pre-processing and experimental settings

The HC18 dataset

The HC18 dataset [van den Heuvel et al., 2018b] contains 999 US images acquired during the various trimesters of the pregnancy, along with the corresponding ground truth of the skull contour map and HC values. The reference contour of fetus head is annotated as ellipse shape by professional sonographer and the HC value as well as pixel size of each image is given in a text file. The gestational age range of this dataset is 10-40 weeks [van den Heuvel et al., 2018b].

Data pre-processing

Image preprocessing includes a resizing from 800×540 pixels to 224×224, and normalization by subtracting the mean and dividing by standard deviation. The HC values are normalized by dividing by the maximum value of HC, in order to improve convergence. We split the dataset into training set (600 images), validation set (199 images) and test set (200 images) in random order. We augment the data of the training set by performing horizontal flipping, and rotation with 10 degrees, the amount of training data is 1800 images.

Experiment configuration

In order to create a fair experimental environment, both approaches, segmentation or regression, are evaluated with the same protocol, namely with 5-fold cross validation, the folds being identical for all the methods. We set the optimizer as Adam with a learning rate of 10^{-4}. The batch size is 16. The training takes 100 epochs. The implementation is based on deep learning framework Keras. In the segmentation experiments, we use the exsisted public Python library Segmentation Models [Yakubovskiy, 2019]. The programs are executed on Tesla P100 GPU server with 16 GB memory.

Evaluation metrics

Evaluation metrics for the segmentation results are the Dice index (DI), the Hausdorff distance (HD), and the Average symmetric surface distance (ASSD). The mean absolute error (MAE) and the percentage MAE (PMAE) are used to compare the predicted and the ground truth HC values.

4.4.2 HC estimation based on segmentation CNN

We train and test 6 different segmentation architectures (U-Net, U-Net++, DoubleU-Net, FPN, LinkNet, PSPNet) with three pretrained CNN backbones (VGG16, ResNet50, EfficientNet). Besides, we added the original U-Net architecture [Ronneberger et al., 2015] that does not have any backbone. We found that the segmentation models pretrained on ResNet50 outperformed the other two CNN backbones. As shown in Table 4.2, that contains both the segmentation accuracy and the HC estimation MAE. From this table, one can gather that:

- Segmentation-wise, all segmentation models obtained similar scores, as shown by values un columns DI, HD and ASSD in the Table 4.2. And these segmentation accuracy have outperformed that in the literature [Budd et al., 2019, Sobhaninia et al., 2019].
- However, when it comes to the estimation error of the HC, the U-Net-B2 and LinkNet-B2 are the best architectures, as assessed by a two-sided, paired Student's t-test between pair of method scores, that resulted in a p-value inferior to 0.05 for these 2 networks. Both networks achieve an MAE value (after post-processing) of 1.08 mm and 1.15 mm respectively.
- Post-processing allows indeed to obtain a small enhancement in the MAE value compared to the results without post-processing.
- Transfer learning techniques help to improve the segmentation accuracy when comparing U-Net with pretrained ResNet50 and U-Net with initial ResNet50.

We also analysed some segmentation results (Figure 4.3) on some vague US fetus head images, the influence of noise and artifacts of images in segmentation-based methods is less than that in the segmentation-free methods (presented in Figure 4.5).

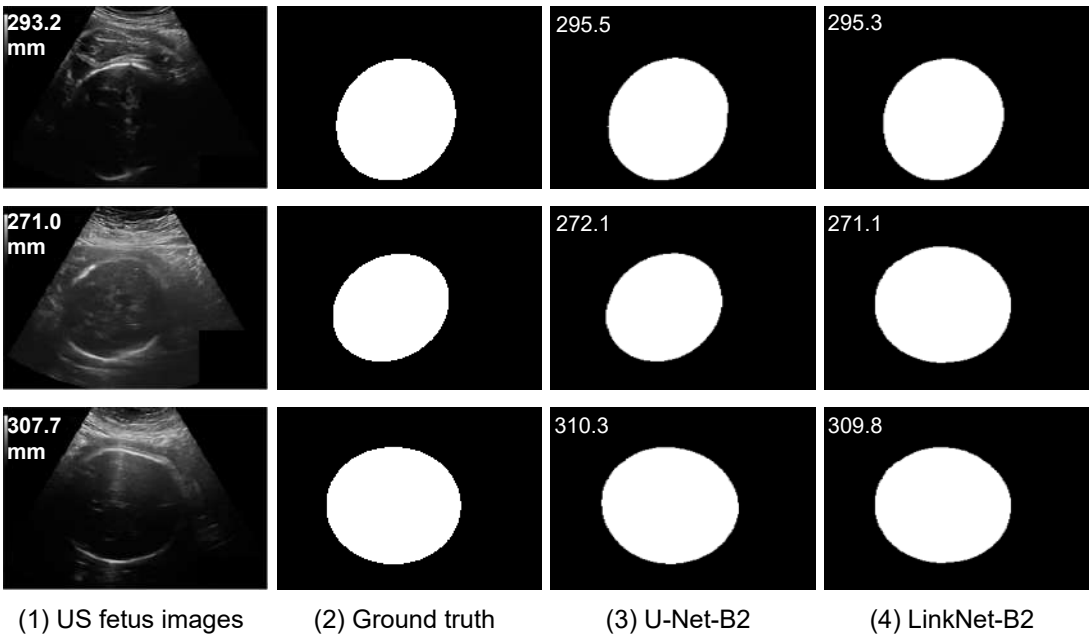


Figure 4.3 – Segmentation results on three large fetus head and vague US images with U-Net-B2 and LinkNet-B2.

Table 4.2 – Segmentation accuracy of the 17 segmentation models and HC estimation accuracy with (w) and without (w/o) post-processing (pp). The results are mean and ± standard deviation. DI = Dice Index, HD = Hausdorff Distance, ASSD = Average symmetric surface distance (mm), MAE = Mean Absolute Error (mm, pixel), PMAE = Percentage MAE. B1 = VGG16, B2 = ResNet50, B3 = EfficientNetb2. Best results are in bold.

Method	**DI ↑ (%)**	**HD ↓ (mm)**	**ASSD ↓ (mm)**	**MAE ↓ (mm) w/o pp**	**MAE (mm) w pp**	**MAE (px) w/o pp**	**MAE (px) w pp**	**PMAE ↓ (%) w/o pp**	**PMAE (%) w pp**
U-Net-original	98.5±1.3	1.56±2.67	0.35±0.29	1.55±4.41	1.23±1.49	11.83±38.75	9.11±10.70	1.04±4.13	0.75±1.04
DoubleU-Net	98.7±1.4	1.14±0.85	0.29±0.26	2.60±1.88	2.59±1.88	18.93±11.53	18.76±10.96	1.57±1.19	1.56±1.17
U-Net(B1)	98.6±1.3	1.16±1.24	0.31±0.26	1.31±2.07	1.21±1.29	9.99±18.72	8.98±8.48	0.85±1.98	0.74±0.75
U-Net-B2-no*	98.3±1.7	1.79±2.58	0.39±0.35	1.82±2.90	1.439±1.70	13.95±21.89	10.49±11.22	0.85±1.98	0.74±0.75
U-Net(B2)	**98.8±0.9**	**1.09±1.11**	**0.28±0.22**	**1.16±1.78**	**1.08±1.25**	**8.69±14.40**	**7.86±7.51**	**0.74±1.46**	**0.65±0.68**
U-Net(B3)	98.7±1.1	1.10±1.03	0.29±0.24	1.34±1.97	1.32±1.67	10.23±16.98	9.94±13.58	0.86±1.62	0.83±1.32
U-Net++(B1)	98.5±2.4	1.29±1.46	0.31±0.25	2.03±8.39	1.3±2.12	16.95±77.93	9.91±18.52	1.51±7.73	0.87±2.33
U-Net++(B2)	**98.7±1**	**1.24±1.66**	**0.29±0.22**	**1.74±6.38**	**1.15±1.59**	**12.65±41.16**	**8.63±12.24**	**1.16±4.65**	**0.72±1.13**
U-Net++(B3)	98.7±1.2	1.17±1.28	0.29±0.25	2.32±11.80	1.19±1.44	19.08±108.01	8.92±11.01	1.57±9.02	0.76±1.21
FPN(B1)	98.6±1.1	1.28±1.68	0.32±0.28	1.44±2.43	1.29±1.61	11.17±22.67	9.70±12.84	0.99±2.58	0.80±1.16
FPN(B2)	**98.7±0.9**	**1.18±1.18**	**0.30±0.23**	**1.38±2.16**	**1.26±1.33**	**10.35±17.99**	**9.18±8.58**	**1.90±9.68**	**0.77±0.87**
FPN(B3)	98.7±1	1.19±1.52	0.30±0.25	1.46±1.92	1.39±1.5	11.09±16.01	10.33±10.53	0.94±1.58	0.86±1.06
LinkNet(B1)	98.6±1.2	1.31±1.53	0.33±0.25	1.46±1.914	1.32±1.44	11.32±16.14	9.91±10.23	0.98±1.70	0.83±1.02
LinkNet(B2)	**98.7±1.1**	**1.12±0.99**	**0.30±0.22**	**1.19±1.56**	**1.15±1.32**	**8.86±11.83**	**8.45±8.38**	**0.74±1.08**	**0.69±0.77**
LinkNet(B3)	98.6±1	1.15±1.04	0.31±0.26	1.37±1.94	1.29±1.51	10.55±15.97	9.70±9.84	0.89±1.62	0.79±0.85
PSPNet(B1)	98.6±1.4	2.00±3.88	0.38±0.44	3.07±12.89	1.33±1.38	22.39±79.36	9.84±9.03	2.21±8.94	0.81±0.81
PSPNet(B2)	**98.8**±0.9	1.42±2.31	**0.31±0.27**	1.66±3.62	**1.20±1.34**	11.98±22.70	**8.75±7.98**	1.07±2.47	**0.72±0.68**
PSPNet(B3)	98.7±1.1	**1.12±1.13**	0.32±0.25	**1.38±1.95**	1.29±1.35	**10.59±16.40**	9.64±9.14	**0.93±1.94**	0.81±0.86

* U-Net-B2-no: not pre-trained on ImageNet.

4.4.3 HC estimation based on regression CNN

We train and test regression CNN architectures with 7 different pretrained CNN backbones, experimented with 3 regression loss functions (MAE loss, MSE loss and Huber loss) on the HC18 dataset. The evaluations of direct HC estimation are given in Table 4.3. One can find that the Regression EfficientNet (Reg-B3-L1) in conjunction with the MAE loss, performs better than the other CNN models: the resulting MAE for this regression network is 1.83 mm. This error is not only smaller than the error (1.90mm) based on segmentation methods in the literature [Budd et al., 2019], but also much smaller than the error in manual measurements intra (7 mm) and inter-variability (12 mm) of sonographers.

Table 4.3 – Average performance of 21 regression CNN models over 5 fold cross validation. The results are mean and ± standard deviation. MAE = Mean Absolute Error, PMAE = Percentage MAE. B1 = VGG16, B2 = ResNet50, B3 = EfficientNetb2, B4 = DenseNet121, B5 = Xception, B6 = MobileNet, B7 = InceptionV3, L1 = MAE loss, L2 = MSE loss, L3 = Huber Loss.

Model	**MAE(mm)**	**MAE(px)**	**PMAE(%)**
Reg-B1-L1	3.04±2.97	22.41±19.94	1.94±2.19
Reg-B2-L1	3.24±3.31	24.11±22.65	2.14±2.61
Reg-B3-L1	**1.83±2.11**	**13.57±13.53**	**1.17±1.43**
Reg-B4-L1	12.59±12.49	93.63±83.53	8.68±11.25
Reg-B5-L1	2.96±2.79	22.39±19.34	1.89±1.97
Reg-B6-L1	3.23±3.29	24.29±22.11	2.13±2.50
Reg-B7-L1	3.34±3.49	26.04±27.89	2.28±2.99
Reg-B1-L2	3.16±3.28	23.83±23.13	2.13±2.69
Reg-B2-L2	3.73±3.48	28.41±26.99	2.55±3.15
Reg-B3-L2	**2.35±2.74**	**17.32±17.95**	**1.53±2.02**
Reg-B4-L2	5.69±5.92	43.54±44.89	3.87±4.97
Reg-B5-L2	3.12±3.07	23.77±22.19	1.99±2.27
Reg-B6-L2	4.68±4.17	35.39±30.59	3.10±3.36
Reg-B7-L2	4.33±4.67	32.29±32.60	2.87±3.78
Reg-B1-L3	3.37±3.72	25.75±26.36	2.33±3.05
Reg-B2-L3	3.12±2.97	24.03±23.69	2.11±2.66
Reg-B3-L3	**2.78±3.03**	**20.62±20.22**	**1.79±2.13**
Reg-B4-L3	9.15±9.07	70.49±67.38	6.20±7.39
Reg-B5-L3	3.40±3.09	26.08±21.34	2.19±2.28
Reg-B6-L3	4.30±4.44	32.48±32.45	2.86±3.67
Reg-B7-L3	6.29±13.86	48.39±111.02	4.33±11.25

4.4.4 Interpretability of regression CNN

Because we can see the segmented results directly from segmentation models, and the HC is calculated according to the fitted ellipse, thus the results are trustable. However, contrary to segmentation models, regression models come at a cost of low interpretability, i.e. the model is not providing explicit explanations along with the HC prediction.

In order to shed the light on what is indeed learnt by the regression CNN, we use a post-hoc explanation method to analyse the regression model. In our previous work [Zhang et al., 2020c], we showed that the Layer-wise Relevance Propagation (LRP) method [Bach et al., 2015] was appropriate to explain CNN regression models for this application. The idea of LRP is to compute a relevance score for each input pixel layer by layer in backward direction. It first forward-passes the image so as to collect activation maps and backpropagates the error taking into account the network weights and activations, yielding saliency maps [Morch et al., 1995], in which the areas that most contributed to a decision are highlighted. Note that in [Dobrescu et al., 2019], authors also used LRP method to explain the results of a regression CNN that aims at counting leaf on plant photographs. We agree that Class Activation Map methods such as Grad CAM [Selvaraju et al., 2017] may be interesting since they provide promising human-interpretable visual explanations for a given CNN architecture. Their principle is to using a global average pooling layer, and to compute the saliency map as the weighted combination of the resulting feature maps at the second last (before softmax) layer. Since we do not have classes here but regressed values, it might be interesting to explore "regression activation map" as defined in [Wang and Yang, 2018].

One can discover from Figure 4.4 that the regression CNN can indeed find the key features from head contour on the input US images and relies on, to some extent, on many contour pixel to make the HC estimation. This indicates that the predictions of regression CNN are reliable to some extent. On the other hand, we can tell from Figure 4.4 that the feature extraction capability of each regression CNN is different according to the red contribution points.

We also display some saliency maps where regression models fail to make an accurate estimation (see Figure 4.5). We observe that the features extracted by regression CNN models are fooled by hypersignal (i.e. high intensity pixels) above the head, which leads to increased predicted HC values. This illustrates the case where the background is heterogeneous and makes it difficult for the network to distin-

guish the head contour and thus to accurately estimate the head circumference. For more analysis on explainability of regression CNNs, please refer to Appendix A.

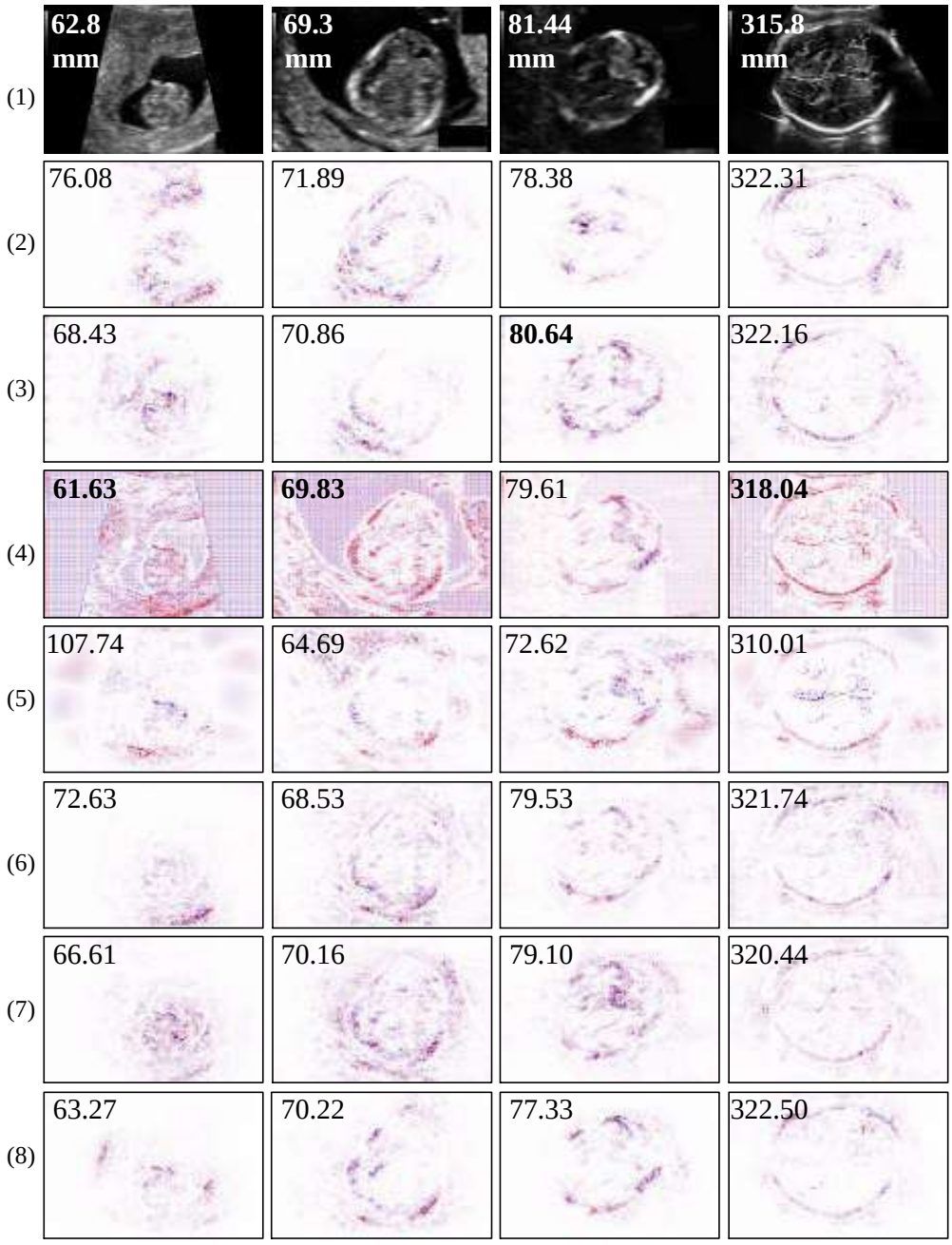


Figure 4.4 – Saliency maps of different regression CNNs explained by LRP method. Row (1), the input US fetus images (numbers are the ground truth HC values.); (2), Regression VGG16 (numbers are the predicted HC values.); (3), Regression ResNet50; (4), Regression EfficientNetb2; (5), Regression Densenet121; (6), Regression Xception; (7), Regression Mobilenet; (8), Regression InceptionV3. The best predicted results are in bold. The red points in saliency maps are positive values, the blue points are negative values.

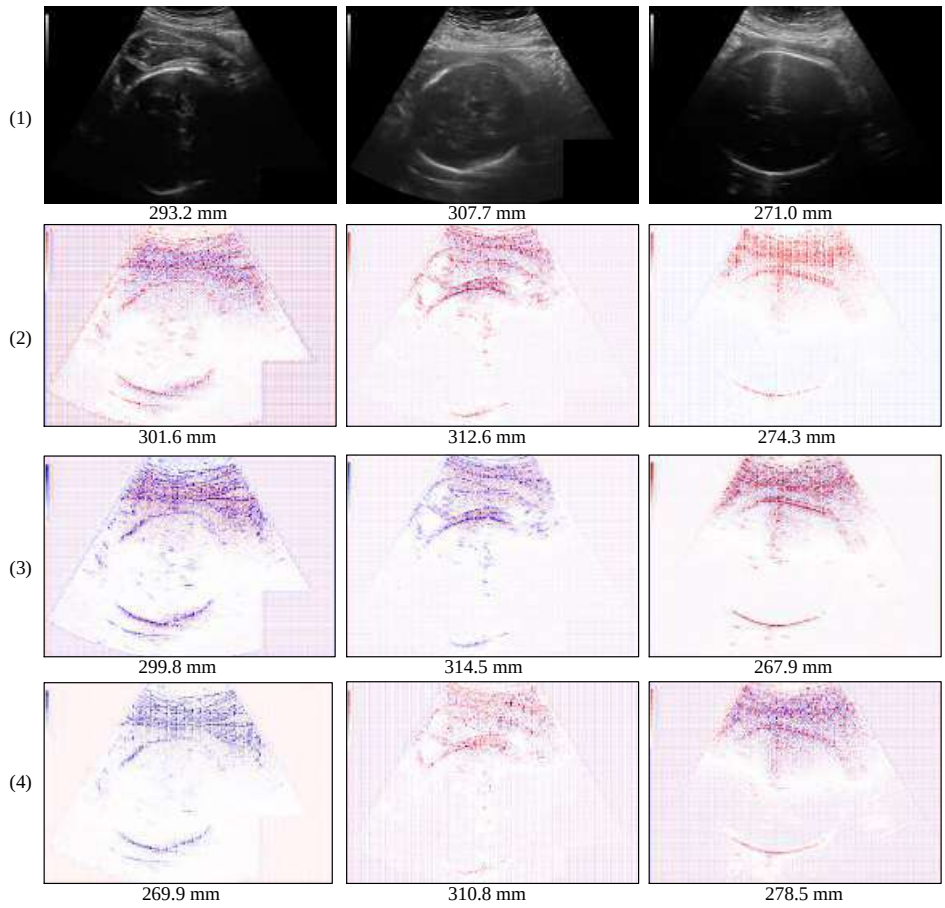


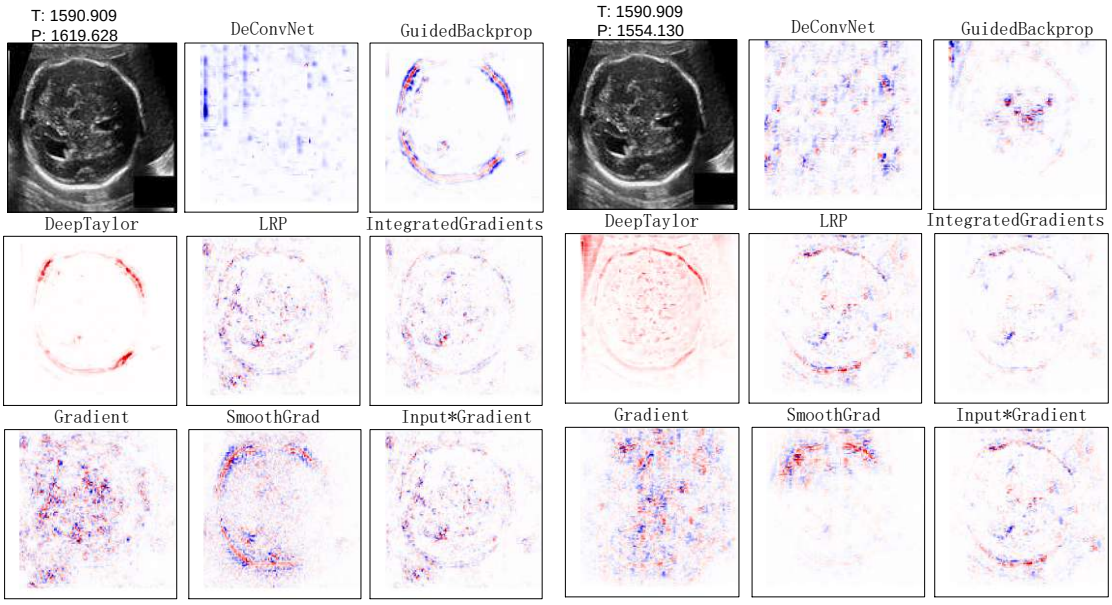
Figure 4.5 – Saliency maps of regression CNN models on 3 cases of bad prediction results. (1) Input fetus head US images and ground truth HC values; (2) Regression EfficientNet with MAE loss and predicted HC; (3) Regression EfficientNet with MSE loss and predicted HC; (4) Regression EfficientNet with Huber loss and predicted HC. Red points means positive contribution, blue points means negative contribution.

4.4.5 Evaluation of explanation methods

Qualitative evaluation of explanation methods

We visualize the saliency maps provided by the 8 selected explanation methods in Figure 4.6. From these images, we can barely see the features retrieved by explanation method DeConvNet and Gradient in both models, that is to say these two methods seem somehow insensitive to the models. This may be explained by the gradient shattering problem [Balduzzi et al., 2017] for the gradient method. Regarding DeConvNet's saliency map, it may be due to the the architecture of deconvolution network which reconstructs the convolution networks reversely. In addition, for

Reg-ResNet50, methods Gradient, GuidedBackprop and SmoothGrad fail to highlight the head contour. We will see that these observations are confirmed by the quantitative evaluation.



(a) Reg-VGG16 (b) Reg-ResNet50

Figure 4.6 – Comparison of different saliency maps with Reg-VGG16 and Reg-ResNet50. P: predicted HC value, T: ground truth HC value (in pixels).

Quantitative evaluation of explanation methods

Here, we compare the explanation methods through perturbation analysis. In this experiment, the input image of size 128×128 pixels is divided into a grid of 4×4 subwindows of size 32×32 pixels. Gaussian noise with mean value 0 and standard deviation 0.3 is added to each subwindow, according to their importance assigned by analyzers during the 16 steps. The input data is the test data set (200 images) and corresponding ground truth. Thus it is slightly time consuming than the saliency map. Figure 4.7 is an example of the perturbation process of Gradient analyzer.

In Figure 4.8, we show the evolution of the prediction error w.r.t. the quantity of noise added at each perturbation steps, on first the most significant subwindow in the analyzer's sense, to the least significant one. One can observe that consistently, the prediction error is increasing, as the level of noise increases. Methods with the steepest curve, LRP and Input*gradient, exhibit the largest sensitivity to perturbations, and as such, should highlight the contributing pixels, in the sense

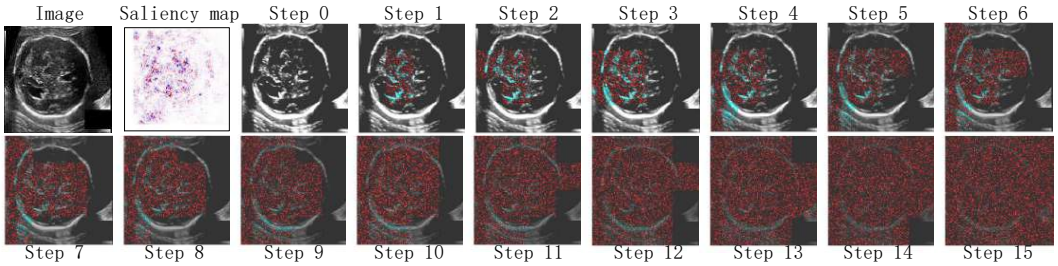
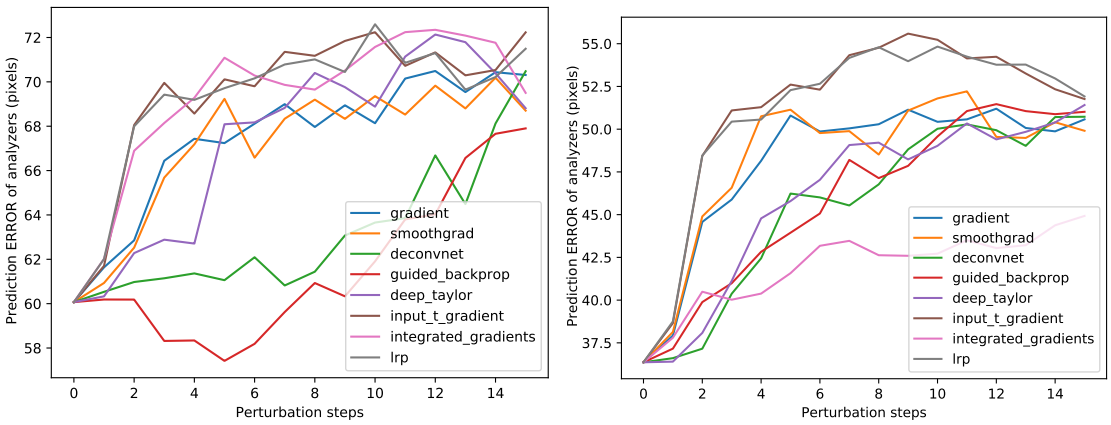


Figure 4.7 – Perturbation process for the saliency map produced by the Gradient method. Step 0 is the original input image. From step 1 to step 15, Gaussian noise is added gradually on the image subwindows. The perturbation order of these subwindows corresponds to the saliency scores assigned by the Gradient method analysis, i.e. the most contributing pixels are perturbed first. Red: noise, blue: original image pixels.

of this criterion. Interestingly the Integrated gradient analyzer seems to be relevant for VGG16, but not for Reg-ResNet50. In the future, it will be interesting to vary the subwindow size to see if results are affected. We expect that a finer grid will be better suited to a thin structure like the head skull. We adapted the evaluation metric of the regression CNN model from accuracy to predicted error, thus, with the noise added gradually according to the importance, the loss increases, the faster the loss increase, which means the better analyzer it is.



(a) Reg-VGG16 (b) Reg-ResNet50

Figure 4.8 – Prediction error (in pixels) of different analyzers during each perturbation step based on Regression VGG16 and Regression ResNet50 model. The horizontal axis is the perturbation steps.

If the relevant features are blocked by Gaussian noise, the model can not predict well, then the analyzer can not detect the relevant feature neither. Therefore, the sensitive analyser will drop steeply, while the insensitive analyser does not change a

lot. To quantify this criteria, the area of perturbation curve (AOPC, Equation 4.5) is regarded as score of one explanation method, for instance, in Figure 4.7, the AOPC value is the difference between accuracy of input image and the average accuracy of 15 perturbed images. Different from classification or segmentation CNN models, the evaluation metrics of regression CNN model is the loss between ground truth and predicted value, not the accuracy, thus we convert the loss by adding a negative sign. In Table 4.4, we compared AOPC scores on regression VGG16 and regression ResNet50 models respectively. Since the AOPC is the difference between the prediction error with and without perturbation, we expect that the analyzer that are indeed perturbed by the noise will return a large AOPC score, in absolute value. We can see that the regression ResNet50 has higher AOPC score than regression VGG16 model. Again we can gather from this table that both the LRP and Input*Gradient methods perform well in those two models. Note that other explanation methods have inconsistent performance depending on the model. This highlights the necessity to choose the proper explanation method before analyzing a specific model.

Table 4.4 – Performance (AOPC scores) of different explanation methods after perturbation, with two regression models. G: Gradient, SG: SmoothGrad, DCN: DeConvNet, DT: DeepTaylor, GB: GuidedBackprop, I*G: Input*Gradient, IG: IntegratedGradients. Lower is better. Best scores in bold.

Model	**G**	**SG**	**DCN**	**DT**	**GB**	**I*G**	**IG**	**LRP**
Reg_VGG16	-7.31	-7.39	-2.87	-7.40	-1.66	**-9.19**	**-9.49**	**-9.17**
Reg_ResNet50	-11.53	-11.84	-9.25	-9.89	-9.72	**-14.75**	-5.60	**-14.58**

In our head circumference estimation study, we use the selected explanation method to analyse different regression CNNs. On the one hand, the model's explainability can be proved that the direct HC prediction using regression CNN is reasonable and effective. On the other hand, different regression CNN models can be compared in the aspect of feature extraction ability. For more information about the experiments of explainability of regression CNNs, please refer to Appendix A.

4.4.6 Agreement analysis of segmentation CNN vs. regression CNN

Comparison of HC estimation accuracy

To compare the performance of the segmentation-free vs. the segmentation approaches, we have gathered the 2 best results from Tables 4.2 and 4.3 into Table 4.5.

From this table, one can see that the best segmentation approach (U-Net-B2: U-Net with pretrained ResNet50 with post-processed segmentation results) is better than the best regression approach (Reg-B3-L1) by 40.7%.

Table 4.5 – Comparison of HC estimation for the 2 best segmentation and regression models. B2: Resnet50. B3: EfficientNet, L1 = MAE loss, L2 = MSE loss. The results are mean and ± standard deviation. MAE = Mean absolute error, PMAE = Percentage MAE. The best results are in bold. (p value<0.05)

Metrics	**MAE(mm)**	**MAE(px)**	**PMAE(%)**
Methods		Segmentation-based methods	
U-Net-B2	**1.08±1.25**	**7.87±7.51**	**0.65±0.68**
LinkNet-B2	1.15±1.32	8.45±8.39	0.69±0.77
		Segmentation-free methods	
Reg-B3-L1	**1.83±2.11**	**13.57±13.53**	**1.17±1.43**
Reg-B3-L2	2.35±2.74	17.32 ±17.95	1.53 ±2.02

Comparison of learning curves

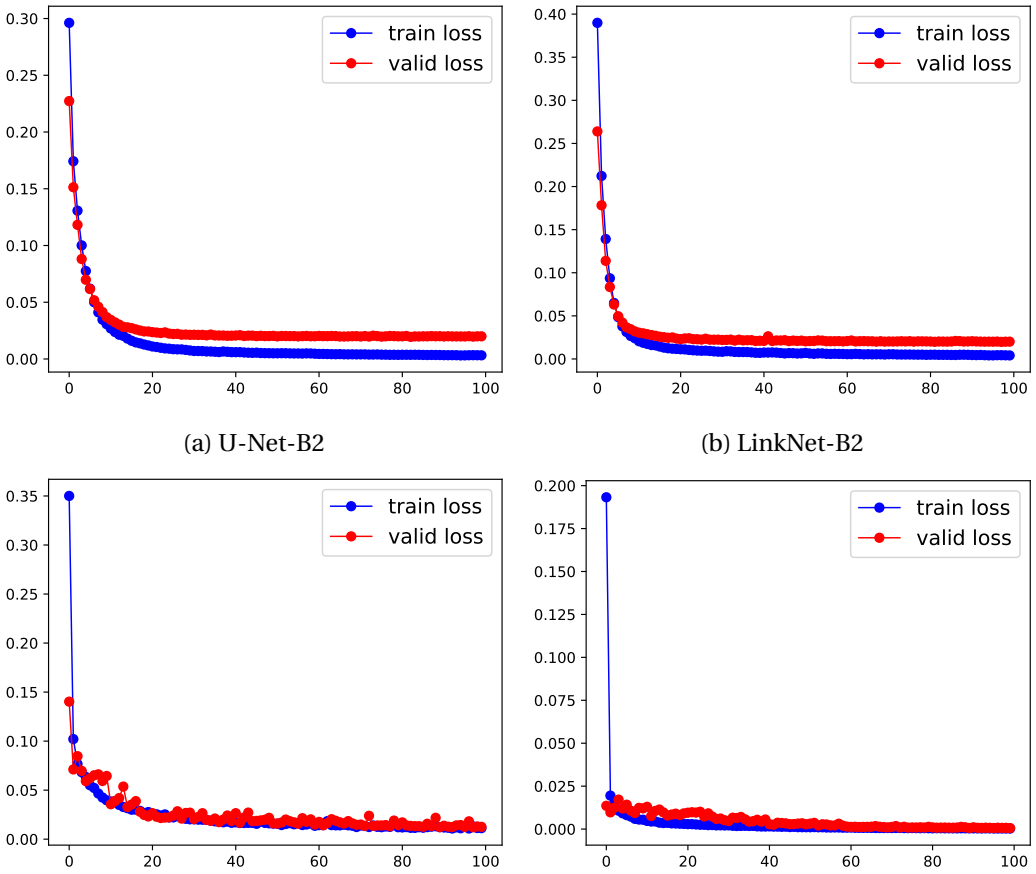
We can also notice from Figure 4.9 that both segmentation and regression methods are correctly fitting the data during training and validation stages, the fitting of the segmentation-based method being even smoother.

Agreement analysis of prediction results

We also analyse the agreement between the estimated HC values by both types of methods against the real HC values via linear regression. From Figure 4.10, one can first observe a remarkable linear correlation between the prediction and the reference values, for all 4 models, whether it is segmentation or regression models. There is a tiny fluctuation in regression CNN models in the right top which illustrates that the regression models have a tendency to underestimate the large HC values (this trend will also appear in the Bland-Altman analysis).

Bland-Altman plot analysis

The Bland-Altman plot is another way to analyze the agreement between two measurements, by plotting the difference between the measurements vs their mean, that makes it easy to spot a bias between the measurements. From the Bland-Altman plot in Figure 4.11, obtained on a fold of 200 test images, we observe that



(a) U-Net-B2

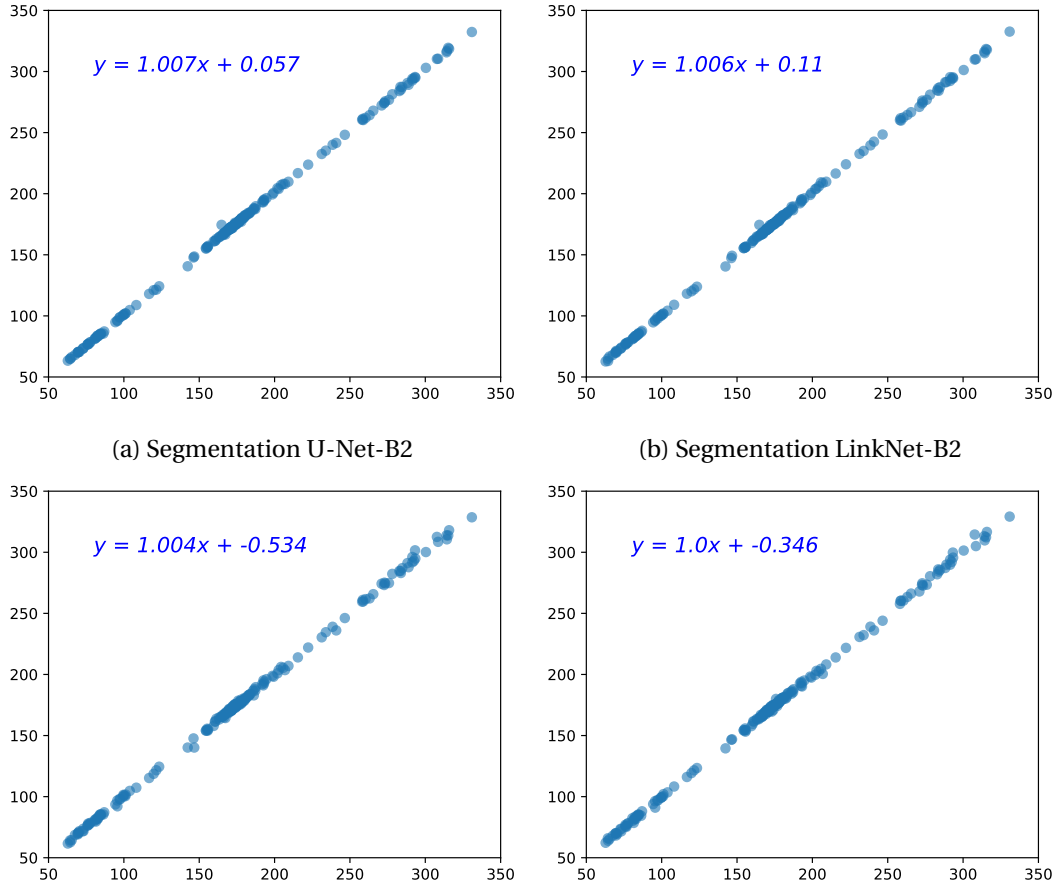
(b) LinkNet-B2

(c) Reg-B3-L1

(d) Reg-B3-L2

Figure 4.9 – Learning curves of segmentation (U-Net-B2, LinkNet-B2) vs. segmentation-free method (Reg-B3-L1, Reg-B3-L2) in training and validation stage. The x-axis represents the training epochs; the y-axis is the loss.

regression approaches struggle with larger fetus head images, which is interesting since segmentation approaches usually fail on small structures. One can also see that for the segmentation models, 8 out of 200 points are outside the 95% agreement limit; for regression models, there are 12 outliers out of 200, mostly distributed in larger HC values. Unsurprisingly, room for improvement is left for regression-based approaches. One can also identify the 95% agreement limits: for the best segmentation model, they are [-3.12mm, 0.7mm], and for the best regression model, they are [-3.25mm, 2.92mm]. We can compare these limits to the 95% agreement limits on inter-operator variability, which is ±12 mm [Sarris et al., 2012, Table 1 page 272]: the fact that they are greatly smaller highlights the high relevance of both of segmentation-based and segmentation-free approaches as alternative to automat-



(a) Segmentation U-Net-B2

(b) Segmentation LinkNet-B2

(c) Regression Reg-B3-L1

(d) Regression Reg-B3-L2

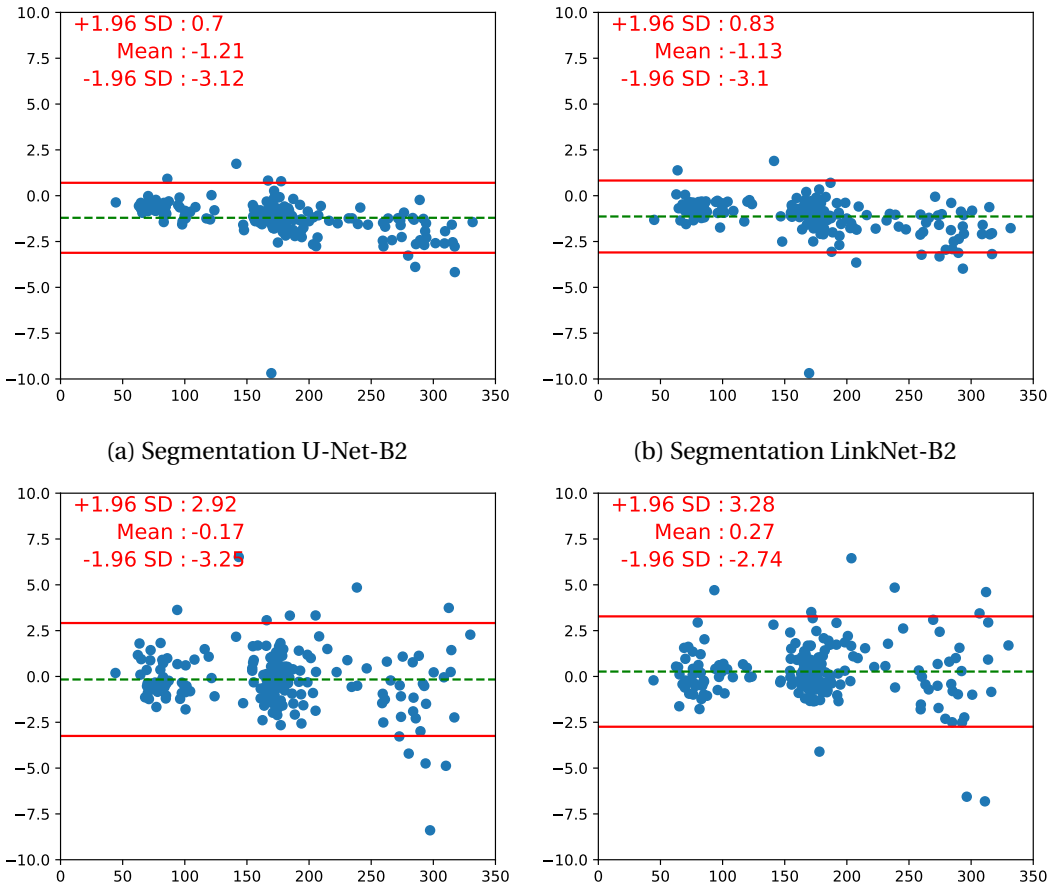
Figure 4.10 – Scatter plots of the 2 best segmentation models U-Net-B2 and LinkNet-B2, and regression models (L1 = MAE loss, L2 = MSE loss). The x-axis represents the ground truth HC and the y-axis the predicted HC (in mm).

ically estimate the HC from US images. However, the comparison to manual variability should be handled with care as these results have not been obtained on the same dataset.

4.4.7 Memory usage and computational efficiency

In addition to prediction accuracy, we also compared the memory usage and computational efficiency of both segmentation-based and segmentation-free approaches. In the aspect of memory usage of a CNN model in theory, Algorithm 4.1[4] gives the pseudo code of estimating memory cost of a model. It simply consists of

[4]The source code reference of computing the theoretical memory of a model:
`https://gist.github.com/jizhang02/ef8eb45450f3d943fea37c6544d3808c`



(a) Segmentation U-Net-B2

(b) Segmentation LinkNet-B2

(c) Regression Reg-B3-L1

(d) Regression Reg-B3-L2

Figure 4.11 – Bland–Altman plots of the segmentation and regression CNN models. The x-axis represents the average value of ground truth and predicted HC; the y-axis, the difference between ground truth and predicted HC (in mm). The horizontal red solid lines represent the upper and lower limits of 95% consistency. The middle dotted green line represents the mean of the difference.

three parts: the memory of embedded model, the memory of model layers as well as the memory of model weights.

Theoretical memory usage of CNN models

The theoretical memory usage of a CNN during training requires to store the network parameters and the activation outputs of every layer, used to compute the gradients, for each batch. As show in Table 4.6, as one could expect, regression CNN models requires less memory storage in general, than the segmentation-based approaches, see column Mem-M. However in practice, the gap between regresssion

Algorithm 4.1 The estimated memory cost of a model

```
Input: Regression CNN Model, batch_size.
Output: Total memory (gigabytes).
procedure MEMORY_USAGE(Model, batch_size)
    shapes_mem_count = 0
    internal_model_mem_count = 0
    for layer_i in Model.layers do
        if layer_type == 'Model' then                          ▷ Embedded model
            internal_model_mem_count +=
            MEMORY_USAGE(batch_size, layer_i)
        end if
        single_layer_mem = 1
        out_shape = layer_i.output_shape                       ▷ Model layers
        for s in out_shape do
            single_layer_mem *= s
            shapes_mem_count += single_layer_mem
        end for
    end for
    trainable_count =SUM(model.trainable_weights)              ▷ Model weights
    non_trainable_count = SUM(model.non_trainable_weights)
    number_size = 8.0                                  ▷ One byte has 8 bits in float 64.
    total_memory = number_size*(batch_size*shapes_mem_count + trainable_-
count + non_trainable_count)
    gbytes = total_memory/(1024^3)+ internal_model_mem_count
    return gbytes
end procedure
```

and segmentation models is not so large, as shown by the actual memory cost in the prediction stage, defined as the maximum used memory when the inference is stable (computed using Python library *Memory Profiler*). In particular, the best regression method (Reg-B3-L1) is even requiring more memory than segmentation methods.

Computational efficiency

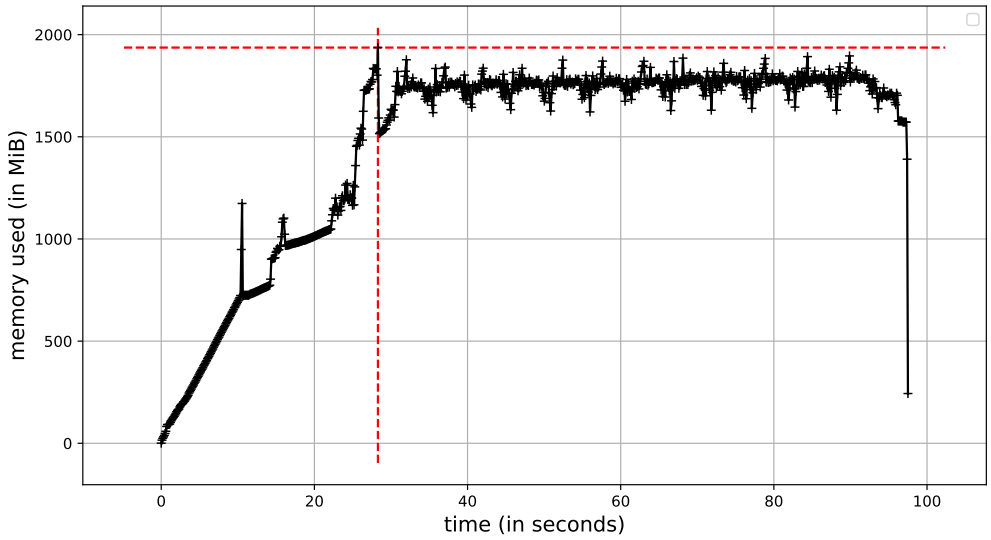
As Table 4.6 shows, the training time per epoch over 1800 training US images for the segmentation method U-Net-B2 (U-Net with ResNet50), takes 29 seconds on a Tesla P100 GPU. For the best regression model Reg-B3-L1 (EfficientNet), it takes 20 seconds. In the prediction stage with a Intel Core i7 CPU, 32 GB RAM, the Regression Reg-B3-L1 only takes 36.95 seconds over 200 test images; in other words,

Table 4.6 – Training and predicting time and memory cost of segmentation vs. segmentation-free models on test set (200 images). B1 = VGG16, B2 = ResNet50, B3 = EfficientNetb2, B4 = DenseNet121, B5 = Xception, B6 = MobileNet, B7 = InceptionV3, L1 = MAE loss, Mem-M= theoretical memory of model, Mem-P= memory in prediction stage, GB = gigabyte.

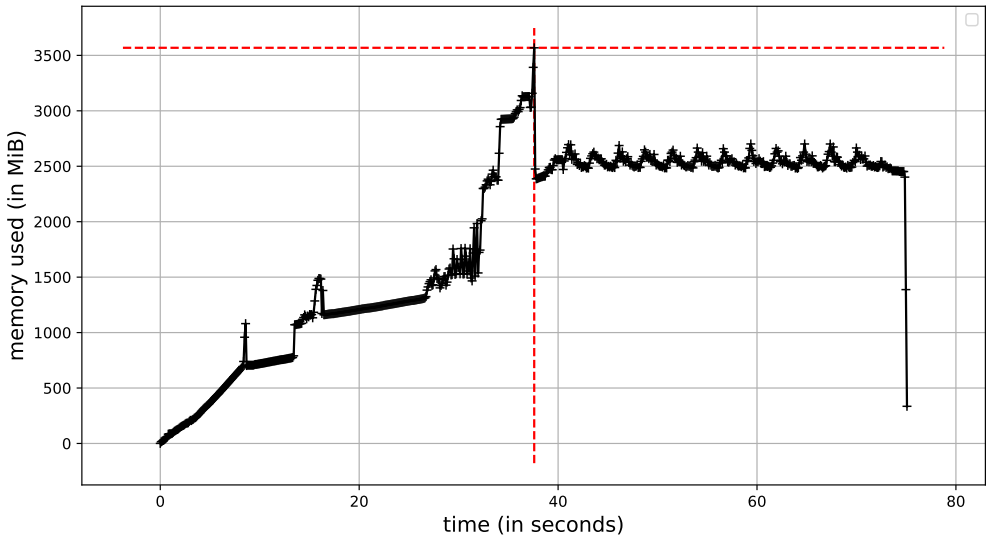
Methods	**Train (s/epoch)**	**Predict (s/test set)**	**Mem-M (GB)**	**Mem-P (GB)**
Segmentation-based methods				
U-Net-B2	29	68.26	3.06	1.84
DoubleU-Net	70	114.21	7.21	2.40
U-Net++-B2	68	172.45	7.26	2.34
FPN-B2	44	101.30	5.47	2.04
LinkNet-B2	30	80.36	3.82	1.90
PSPNet-B2	88	225.38	11.06	4.04
Segmentation-free methods				
Reg-B1-L1	17	30.86	0.96	1.36
Reg-B2-L1	20	48.28	2.31	1.73
Reg-B3-L1	38	36.95	2.29	2.68
Reg-B4-L1	21	65.55	3.01	1.69
Reg-B5-L1	35	51.78	2.15	1.67
Reg-B6-L1	14	18.71	1.03	1.14
Reg-B7-L1	17	22.55	1.09	1.60

predicting one image requires 0.18 second, to be compared to 0.35 seconds of the U-Net-B2. Segmentation-based methods require longer time at training but also at inference time, than segmentation-free methods. As a conclusion, whereas the advantage of using regression-based approach is clear computationwise, there is no clear evidence that regression models are less memory greedy, in the experimental conditions we set up. It's worthy to note that with the continuous progress of hardware and computing power, such time error between segmentation-based and segmentation-free methods may be ignored in clinical practice.

Figure 4.12 compare the best segmentation-based and segmentation-free methods in terms of memory cost during the model prediction stage. From the figure, we can find that predicting the same number of images, the segmentation-free method takes less time than the segmentation method. This is because there are two processes in the segmentation model, image feature extraction (Encoder) and feature upsampling (Decoder).



(a) Memory cost of U-Net-B2



(b) Memory cost of Reg-B3-L1

Figure 4.12 – Memory cost of (a) segmentation (U-Net, B2=ResNet) vs. (b) segmentation-free method (Regression B3=EfficientNet, L1=MAE) during prediction stage. The maximum used memory of models (when the inference is stable) is after the red vertical line.

4.4.8 Comparison of HC estimation with state-of-the-art

At last, the proposed segmentation-based methods and segmentation-free methods are compared with state-of-the-art (SotA) methods (Table 4.7). In the SotA solutions, segmentation intervention are still needed although their models are fancy. For example, in [Fiorentino et al., 2021], 3 steps including fetus head localization, segmentation, ellipse fitting are performed for computing HC, which seems to be

cumbersome. While the proposed two kinds of methods can effectively estimate HC through simple architectures such as U-Net with ResNet or regression EfficientNet with the benefit of transfer learning. One should also note that the sota methods are not that comparable with each other due to their different experiment protocols. For instance, in [Budd et al., 2019], they trained HC18 dataset combined with other fetus head US images.

Table 4.7 – Comparison of HC estimation with state-of-the-art on HC18 dataset. B2=ResNet50, B3=EfficientNetb2, L1=MAE loss, DI=Dice Index, N/A=Not applicable.

Metrics	**MAE(mm)**	**DI(%)**
Segmentation-based methods		
[Sobhaninia et al., 2019]	2.12±1.87	96.84±2.89
[Budd et al., 2019]	1.81±1.65	98.20±0.80
[Fiorentino et al., 2021]	1.90±1.76	97.75±1.32
[Moccia et al., 2021]	1.95±1.92	97.90±1.11
U-Net-B2(Proposed)	**1.08±1.25**	**98.80±0.9**
Segmentation-free methods		
Reg-B3-L1(Proposed)	**1.83±2.11**	N/A

4.5 Conclusion and future work

In this work, we have addressed the problem of HC estimation from US images via both a conventional segmentation approach with post-processing and ellipse fitting, and a regression-based approach that can directly predict HC without segmentation intervention. Our idea was to quantify how far regression-based approaches stand from segmentation approaches, when the final task is to estimate a parameter, i.e. a biomarker, from the image. Although segmentation-based methods provide interpretable results for the HC estimation because the segmentation result is visible, they often require dedicated post-processing steps. On the other hand, regression approaches based on CNN are end-to-end, less costly and prone to error and even though they do not offer explicit interpretability, this aspect can be explored using saliency maps for example [Zhang et al., 2020c]. In our study, we have explored both segmentation and segmentation-free approaches with state-of-the-art CNN architectures and backbones. By setting the same experimental conditions, we have proposed a fair, quantitative comparison of these two approaches, in order

to assess if the direct estimation approach is viable for this task. Even though the estimation error is much higher with the regression networks, the results are still promising and in line with inter-operator variability. Therefore, direct estimation, regression-based approaches have a high potential that should be deepened in the future. Whereas we used general-purpose architectures for our regression methods, it will be interesting to investigate customized architecture for this task, and that include attention mechanisms.

In the future work, we will assess the generic regression CNNs on other medical datasets to estimate multiple biomarkers. Besides, we plan to investigate the segmentation-free approaches with other, recent CNN architectures that have higher ability of feature representation, e.g transformer architectures, as well as multi-task learning which combines segmentation branch and regression branch.

Chapter 5

Cardiac multi-structure volume prediction

Contents

5.1 Motivation

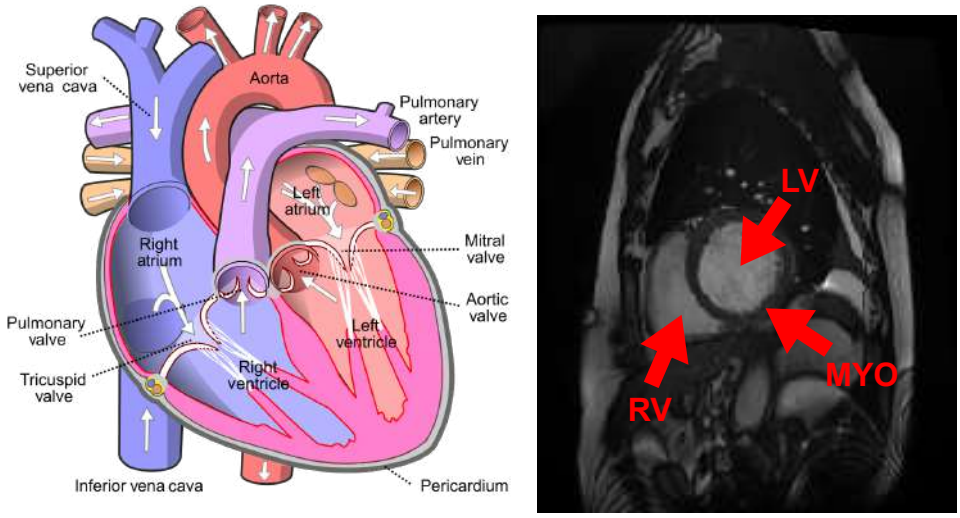
As we have seen in the previous chapter, medical image segmentation is often a prerequisite step toward the computation of biomarkers. In Chapter 3, we tackled the problem of head circumference estimation. In this chapter we focus on another problem that could benefit from direct estimation: estimating the volume of cardiac ventricular cavities and left myocardium from MR images [Petitjean and Dacher, 2011]. This problem is more complex since US images are 2D in the case of head circumference estimation and here MR images are 3D. Morever, in US images, the zone to be segmented, i.e. the skull, is dense and thus appears directly as a contour, whereas in the MR images, the cardiac ventricles and myocardium are in the mediastinum, in the middle of other organs, and their respective boundaries are not especially highlighted.

In this work, we investigate how a vanilla regression CNN can perform to estimate automatic multi-structure cardiac volume without segmentation. The method is performed on public "Automatic Cardiac Diagnosis Challenge" dataset (ACDC), the predicting targets are the volume of RV, LV, and MYO, respectively.

This chapter is organized as follows: in Section 5.2 we present some background in MR image processing for cardiac function evaluation. Section 5.3 we introduce the dataset preprocessing and regression models. Experimental results and limitations are discussed and presented in Section 5.4. And conclusions are drawn in Section 5.5.

5.2 Background on cardiac function evaluation

The cardiovascular diseases (CVDs) are one of most common diseases in the world, which is the leading cause of death globally, taking an estimated 17.9 million lives each year. CVDs are a group of disorders of the heart and blood vessels and include coronary heart disease, cerebrovascular disease, rheumatic heart disease and other conditions. More than four out of five CVD deaths are due to heart attacks and strokes, and one third of these deaths occur prematurely in people under 70 years of age according to the information of World Health Organization [WHO, 2021]. Measuring the volume of cardiac sub-structures is a basic operation for assessing the cardiac function. For instance, the ejection fractions and stroke volumes of left and right ventricular in both end diastolic and end systolic



(a) Diagram of a human heart (b) MR image of a cardiac

Figure 5.1 – The structures of a human cardiac. (a) Diagram of a human heart (The figure is taken from `Wikimedia Commons`); (b) 3 structures (RV, MYO, LV) of cardiac in short axis view from MR image (The figure is taken from ACDC dataset).

phases, as well as the mass of left ventricle. These biometrics are all related to whether a patient's cardiac function is normal or not. In clinical routine, the conventional practice is delineation of cardiac, which is semi-automatic, though, then following step is calculating volume or area based on segmentation results. With the boost of deep learning techniques on medical imaging, the automatic segmentation of cardiac has been promoted to a new level. Common cardiac data format is magnetic resonance imaging (MRI), computed tomography (CT) and ultrasound images. Current main segmentation applications of cardiac are bi-ventricle (left ventricle, LV, right ventricle, RV) and bi-atrium (left atrium, LA, right atrium, RA), coronary artery as well as myocardium (MYO). At last, the doctors make the diagnose based on the indices of different cardiac structures. Figure 5.1 shows the diagram of a human heart as well as a MR image of the cardiac with 3 structures (RV, MYO, LV). Nowadays, there are plenty of researches that focus on cardiac segmentation [Chen et al., 2020a], which involves different types of deep learning techniques, for instance, fully convolutional neural networks (CNN) with 2D or 3D kernels [Jang et al., 2017, Isensee et al., 2017, Yang et al., 2017], generative adversarial networks (GAN) [Savioli et al., 2018], and recent transformers as encoders [Chen et al., 2021b] has further improved the segmentation accuracy, etc. However, for quantifying the segmented results, it still needs two steps (segmenta-

tion+quantification) to obtain the volume of cardiac structures.

Based on the above literature we know that segmentation-based methods are still more popular for the current prediction of the structural volume of the cardiac, but later direct prediction methods have also emerged (See Chapter 2), and among these direct prediction methods for certain biomarkers, there are gradually beginning to be studies for cardiac biomarkers, but the methods for data preprocessing and data augmentation are not very transparent, and there is a lack of interpretable studies of the models, so we have further supplemented and improved this work.

5.3 Methodology

5.3.1 ACDC dataset and preprocessing

The ACDC dataset

The public "Automatic Cardiac Diagnosis Challenge" dataset (ACDC) dataset [Bernard et al., 2018] is used in this study. These data are obtained on two different MRI scanners with different magnetic strengths. Generally, the image quality is better when the magnetic strength is higher. The dataset contains 100 magnetic resonance images (MRI) subjects in training set, each subject has 3 manual annotated labels, i.e., left ventricular (LV), myocardium (MYO), right ventricle (RV). Each subject has end diastolic (ED) and end systolic (ES) phase. Because the machine acquires consecutive frames within one heartbeat cycle. The ED stage and ES stage are selected by an experienced specialist or physician by observing changes in the size of the heart chambers. These subjects are divided into 4 types of disease, myocardial infarction (MINF), dilated cardiomyopathy (DCM), hypertrophic cardiomyopathy (HCM), abnormal right ventricle (ARV), and patients with normal cardiac (NOR). For the ground truth, there are masks of three cardiac structures annotated by experts, and the volumes of RV, MYO and LV. The volume of the cardiac is calculated according to the following formula. In Formula 5.1, N is the number of slice in each subject, *S_RV* is the summary of pixels (area) belong to RV class in one slice (same as *S_MYO, S_LV*), *px_x* (mm) is pixel size of *x* dimension, *px_y* (mm) is pixel size of *y* dimenstion, *space_z* (mm) is the slice thickness. The principle of this formula is to superimpose the area of each slice, so that the cumulative calculated area is the volume of the different structures of the cardiac. For uniformity of units (millime-

ters and milliliters), the result of the calculation is divided by 1000 to turn it into a volume in milliliters.

$$\text{Volume} = \begin{cases} \text{RV} = \sum_{i}^{\text{N}} \text{S\_RV} * \text{px\_x} * \text{px\_y} * \text{space\_z}/1000 \\ \text{MYO} = \sum_{i}^{\text{N}} \text{S\_MYO} * \text{px\_x} * \text{px\_y} * \text{space\_z}/1000 \\ \text{LV} = \sum_{i}^{\text{N}} \text{S\_LV} * \text{px\_x} * \text{px\_y} * \text{space\_z}/1000 \end{cases} \tag{5.1}$$

Data cleaning

Data cleaning is to remove some data that are outliers from the main part, which is necessary before performing certain methods on specific dataset. Otherwise, it will affect the experimental results and accuracy. In ACDC dataset, in order to make sure the ground truth volumes of each subject offered by the authors of ACDC dataset are correct, we check the consistence of given volumes and volumes computed through Formula 5.1. We found that there are 6 patients out of 100 pairs subjects (including ED and ES) whose volumes of cardiac are seriously deviated from the computed volumes. And the difference are up to hundred level which can not be ignored. See Table 5.1. The left subjects in given volumes are consistent with computed volumes (because their differences are less than 1). Thus, we will remove these 6 patients in the experiments.

Table 5.1 – Abnormal samples in ACDC dataset, S is the total pixel numbers of all sllices, V_c is volume from calculation (see Formula 5.1), V_gt is volume from given data, Diff is the difference between V_g and V_c.

Patient	**spacing(x,y,z)**	**shape(z,y,x)**	**S(pixel)**	**V_c(ml)**	**V_gt(ml)**	**Diff(ml)**
P019	(1.445,1.445,10)	(11, 256, 216)	32269	673.78	868.59	194.81
P078	(1.367,1.367,10)	(8, 256, 216)	25813	482.36	630.20	147.84
P079	(1.367,1.367,10)	(9, 256, 216)	18667	348.83	455.74	106.91
P080	(1.758,1.758,10)	(6, 256, 216)	9173	283.50	223.95	59.55
P093	(1.563,1.563,7)	(10, 224, 180)	23491	401.71	57.35	344.35
P099	(1.786,1.786,5)	(16, 224, 154)	27180	433.49	866.71	433.21

Data preprocessing

Image cropping and ROI detection From Table 5.1 we can also see that each original data has different sizes in 3 dimensions. Moreover, the cardiac takes small proportion in each slice (See Figure 5.2). And this is not conducive to the training of convolutional neural networks. Because the model requires the fixed shape of input data and the target/feature should be as clear as possible.
Therefore, we perform data cropping to uniform the shape of 3D MR images and find the region of interest (ROI) of cardiac. We first find the maximum bounding box of the cardiac from ground truth images[1]. Then we crop the MR images based on the maximum bounding box (See Algorithm 5.1).

Algorithm 5.1 Cardiac MRI cropping algorithm

Input: Original Cardiac MRI and ground truth (GT).
Output: Cropped Cardiac MRI.

for subject_i in GT_Dataset **do** ▷ Finding max Bounding box in ground truth dataset.
    Max_BoundingBox = $(x_0, y_0, \Delta x, \Delta y)$
    for slice_k in subject_i **do**
        if BoundingBox_k > Max_BoundingBox **then**
            Max_BoundingBox = BoundingBox_k
        end if
    end for
end for
for subject_i in MRI_Dataset **do** ▷ Cropping in MRI dataset
    for slice_k in subject_i **do**
        slice_k = Crop(⌈ Max_BoundingBox ⌉)
    end for
end for

From Figure 5.2, we can see that the cropped MI image maximizes the retention of the cardiac target and removes other organs or noise from the image, which facilitates the learning of features in the image by the deep model.
Uniforming number of slices After completing the fixation of the cardiac target in the two-dimensional direction, the number of slices should also be consistent for each MRI data, i.e., we added or removed cardiac slices at minimal cost in order to satisfy the principle of constant input data size. Specifically, we select the median of all data depths in the ACDC dataset as the uniform number of slices, and for data

[1] In this step, we remove 3 extreme examples because the size of bounding box are even larger than the size of images of others.

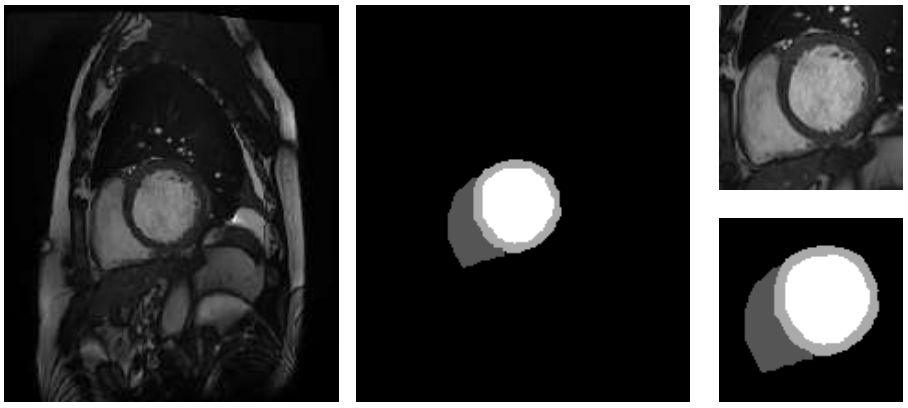


Figure 5.2 – One slice of original ACDC data and ground truth (gt), the size is 182*216, the right column is the cropped MRI cardiac subject and its ground truth, the size is 100*100.

above that number, we remove them from the bottom of the cardiac (because the area of the slices at the bottom of the cardiac is the smallest), and for data below that number, we duplicate the slices at the bottom of the cardiac and accumulate them until we reach that number. At last, the shape of a 3D MR image is $(100 \times 100 \times 9)$.

Data augmentation based on grid search

After the above data preprocessing steps, there are 182 valid images whose shapes are $(100 \times 100 \times 9)$. However, this number of data sets is far from sufficient for a deep neural network model with a large number of learnable parameters and can easily lead to overfitting. Usually, data augmentation is the common way to increase the number of images, specifically through image processing techniques (rotation, translation, etc.) to increase the diversity of images.

In this study, given the small amount of raw data, we develop an efficient automatic data augmentation algorithm. The algorithm is based on a backtracking method to find a subset without duplicates, i.e., this is a grid search to enumerate different combinations of image processing algorithms to add a custom number of images.

First of all, we list 10 different basic image processing algorithms. Then, the index of these functions are sent into the Algorithm 5.3 [2].
①.'aug_rotate'; ②.'aug_rotate_r'; ③. 'aug_flip_h';

[2]The source code is at
`https://gist.github.com/jizhang02/4f4a08aa54fe39e4a0ac9b272562bde4`

Algorithm 5.2 Cardiac MRI slice uniforming algorithm

Input: Original Cardiac MRI.
Output: Cardiac MRI with same number of slices.

```
Gold_Slice = 9
for subject_i in MRI_Dataset do
    Initialize Novoid_subject
    Initialize Minarea
    Initialize Mindex                                   ▷ The index of Minarea
    for slice_k in subject_i do
        if slice_k != NULL then
            Novoid_subject_i = append(slice_k)          ▷ Remove void slice
            if Area(slice_k)<Minarea then
                Minarea = Area(slice_k)
                Mindex = k
            end if
        end if
    end for
    while Number_Slice_Novoid_subject_i < Gold_Slice do
        Novoid_subject_i.append(slice_Mindex)           ▷ Append the Minarea slice
    end while
    while Number_Slice_Novoid_subject_i > Gold_Slice do
        delete Novoid_subject_i[-1]                     ▷ Delete from the last slice
    end while
end for
```

④. 'aug_flip_v'; ⑤.'aug_trans_x'; ⑥. 'aug_trans_y'; ⑦. 'aug_shear_x';
⑧. 'aug_shear_y'; ⑨. 'aug_gauss'; ⑩. 'aug_gamma_correct'.

Having the above 10 single data augmentation methods, we use backtracking method to find all the subsets of these 10 methods, then $2^{10} = 1024$ different combinations (subsets) will be generated. For example, the original dataset has 182 subjects, if the user wants to generate 5000 subjects, then we take 5000/182 = 28 combinations from 1024. This ensures each subject is different from the others. In this work, we generate 28*182 = 5096 synthetic images.

5.3.2 Regression CNN

2D convolution for 3D data

Given that MRI or CT images are in a 3-dimensional format, the third (z direction) dimension is the depth of the organ scan or the number of slices. From the point of

Algorithm 5.3 Data augmentation based on grid search

```
Input: Index of function list.                          ▷ Functions of data augmentation
Output: Subsets without duplicate.
Initialize result                                       ▷ Store all the subsets
Initialize temp_set                                     ▷ Temporal set
procedure BACKTRACKING(nums, startIndex)
    if startIndex >= nums.size then
        return
    end if
    for startIndex in nums.size do                      ▷ Horizontal traversal
        temp_set.append(nums[startIndex])
        result.append(temp_set)
        BACKTRACKING(nums, startIndex+1)                ▷ Vertical traversal
        temp_set = temp_set[-1]
    end for
end procedure
procedure SUBSETS(Index of function list)
    BACKTRACKING(Index of function list,0)
    return result
end procedure
```

view of the image, it can also be considered to be composed of multiple channels, see Figure 5.3.

When dealing these 3D cardiac data with deep convolutional neural networks, it would be natural to think of using a 3-dimensional convolution kernel to learn the neighborhood features and spatial information of that data, for example, the segmentation models 3D U-Net [Çiçek et al., 2016], V-Net [Milletari et al., 2016], etc. In addition to 3D convolution, it is possible to utilize 2D convolutional neural networks on 3D images, in which each slice is regarded as one input channel. The literature [Yang et al., 2021], [Hassanzadeh et al., 2020], [Vu et al., 2020] has proved the feasibility of 2D CNN on multiple image slices.

For 2D convolution, the input layer and the filter have the same depth, in other words, the number of image channels is the same with the number of convolutional kernels/filters. The filter slides in 2D direction. Then the input and the filter are summed together into one feature map, each element is a pixel. Iteratively, the feature map goes deeper with more filters, which depends on the architecture of CNN model.

For 3D convolution, the filter is a 3D kernel, which is generally $(3 \times 3 \times 3)$, the

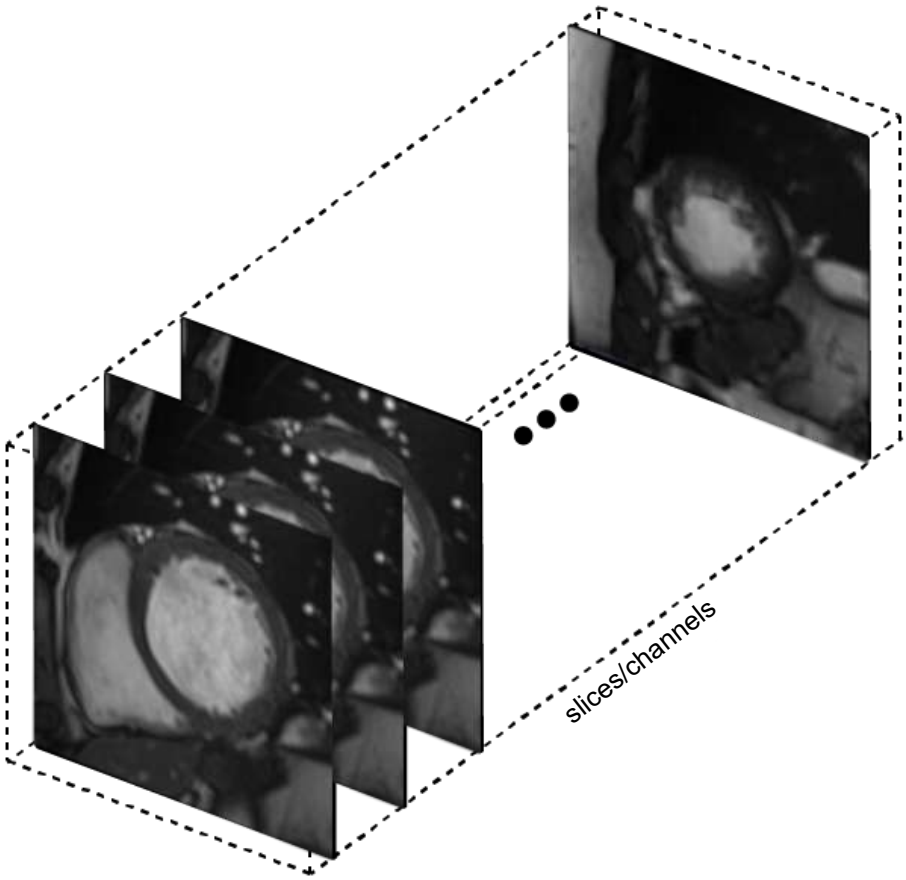
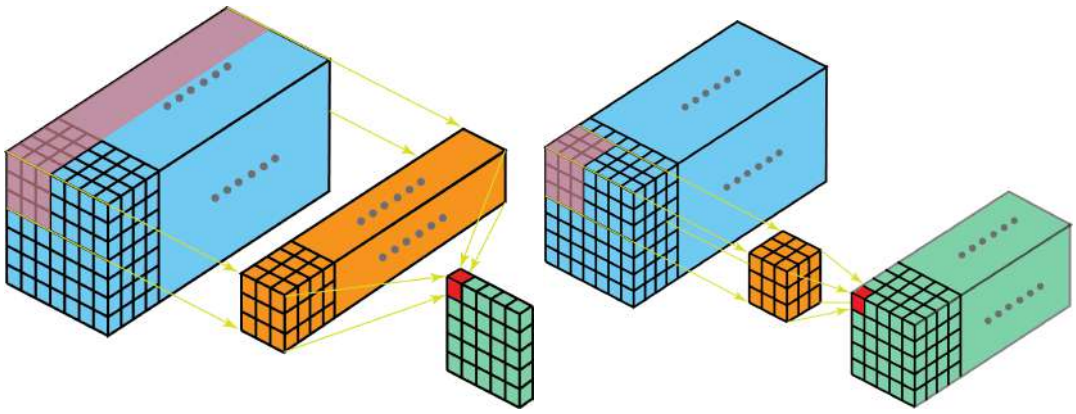


Figure 5.3 – One of preprocessed 3D cardiac data from ACDC dataset. The cardiac slices can be regarded as image channels.

filter moves on 2D channel first, then moves in z direction. The output is a 3D matrix. That is to say, each element is a voxel. Afterwards, the 3D feature map goes into next layers with more 3D filters. Because 3D convolutions can describe the spatial relationships of objects in the 3D space. It is beneficial for some applications, such as 3D segmentation/reconstruction of biomedical imagining, which is related to voxel-wise classification. Figure 5.4 shows the 2D and 3D convolution process on 3D data.

In our study, our task is image-wise regression. That is, the prediction results of the model are determined based on the most important feature of the whole image. Thus, this feature can be learned either by 2D or 3D convolutional kernels. However, 2D convolution and 3D convolution models are much different in terms of time and space complexity, and when combined with the above figure (Figure 5.4), 3D convolution models require a large number of training parameters, and this huge number of parameters requires large memory space and computational power. Table



(a) 2D convolution (b) 3D convolution

Figure 5.4 – 2D (a) and 3D (b) convolution on 3D image. For 2D convolution, the input layer and the filter have the same depth (channel number = kernel/filter number), the output is a one-layer matrix. For 3D convolution, the filter moves on 2D channel then moves in z direction, the output is a 3D matrix. The figure is taken from `Towardsdatascience`.

5.2 compares 2D regression CNN with 3D regression CNN in the aspects of number of parameters, model memory and training time, etc. The backbone is based on VGG16 [Simonyan and Zisserman, 2015]. The convolution kernel of 2D regression VGG is (3×3), $(3 \times 3 \times 3)$ in 3D regression VGG. One can find that the 2D regression VGG takes less time and memory than that with 3D regression VGG. Therefore, considering the above analysis, we will choose 2D regression CNN to predict the volume of different structures of 3D cardiac data.

Table 5.2 – Comparison of 2D regression VGG and 3D regression VGG. The training time is on the GPU server; the model memory is theoretical requirement; the inferencing memory is actual memory cost during the prediction stage of a model; M=Million; GB=Gigabytes; N/A=Not applicable.

Model type	**# of param (M)**	**Training time s/epoch**	**Model memory (GB)**	**Inferencing memory (GB)**
RegVGG_2D	14.72	21	0.518	0.83
RegVGG_3D	44.93	719	3.724	N/A*

* The 5-fold training time of regression VGG 3D (around 100 hours) exceeds the maximum GPU time (48 hours).

The architecture of regression CNN

Regression CNNs We design a deep regression CNN architecture shown in Figure 5.5. Any CNN model can be a backbone to learn the feature from

training dataset, for instance, VGG16 [Simonyan and Zisserman, 2015], ResNetV2 [He et al., 2016b] or EfficientNet [Tan and Le, 2019]. For the regression part, after the feature maps were flatten into fully connected layer, we simply use linear regression on it to predict the volume of 3 structures of cardiac. This model is similar to our previous work on fetus head circumference prediction. The difference lies in the initial input layer and the final output layer. Our work is a multi-structured volume prediction of 3D cardiac data.

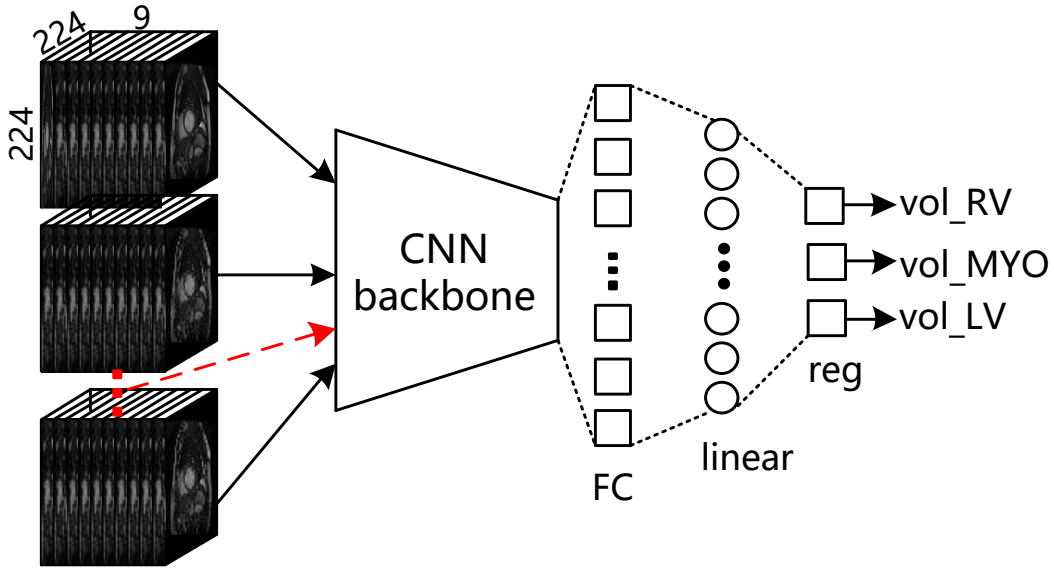


Figure 5.5 – The architecture of regression CNN for predicting the volume of RV, MYO, LV directly. The input training data are preprocessed MRI images (224,224,9), the ground truth are the volume of 3 structures of cardiac. FC is fully connected layer after feature representation, the FC layer goes through linear regression layer, the output is the predicted volumes (vol_RV, vol_MYO, vol_LV).

Loss functions Same as previous work, the regression CNN model is optimized by regression loss function such as mean absolute error (MAE) loss or mean square error (MSE) loss or Huber loss (HL) [Esmaeili and Marvasti, 2019].

Interpretability of regression CNNs Because we can see the segmented results directly from segmentation models, and the cardiac volume is calculated according to the segmented areas, thus the results are trustable. However, the regression CNN models come at the cost of a low interpretability, i.e. the model is seen as a black box, which does not provide explanations along with the cardiac volume prediction. To this end, we use a post-hoc explanation method to analyse the regression model, in our previous work [Zhang et al., 2020c], we validated that the method Layer-wise

Relevance Propagation (LRP) [Bach et al., 2015] can well explain the regression CNN models in the form of saliency maps [Morch et al., 1995].

Transfer Learning from RGB to multi-channel cardiac MRI data

In order to further improve the model's performance, we use transfer learning strategy to load CNN backbones that are pretrained on ImageNet [Deng et al., 2009]. Although ImageNet images and MR images have obvious dissimilarities, some generic representations can be learnt from a large-scale dataset, that might be beneficial to other types of images, and they have proven so in the context of brain MR images [Wacker et al., 2020]. Since these pre-trained models are trained on natural RGB images, the input depth of the models is 3 channels, which will not work in multi-slice data. To solve this problem, we loop through the layers of the pre-trained CNN model and replicate the average of the existed weights to new channels in each layer so that it can ensure the input layer matches the subsequent layers.

5.4 Experiments and results

5.4.1 Experiment protocol

The ACDC [Bernard et al., 2018] dataset in this study has 182 subjects after data preprocessing. We split it into the training set (100), the validation set (32), the test set (50). The total training set has 2900 3D MR images including data augmentation. The optimizer is Adam. The learning rate is $1e^{-4}$, the batch size is 16. The algorithm is completed using Python and Keras library with GPU p100[3]. The training epoch is 100. 5-fold cross validation is performed. The structure of regression CNN is: 2D convolutional kernel with multi-class volume prediction. For example, the input shape is (N,224,224,9), N is the number of input images, the image is resized to 224*224, one image has 9 slices. In the experiments, we train three different regression CNN backbones, which are VGG16 [Simonyan and Zisserman, 2015], ResNet50V2 [He et al., 2016b] and EfficientNet [Tan and Le, 2019] respectively. These pretrained models originally only had three channels, in this work, we expand these three channels into 9 channels in order to match the data shape. We perform data normalization both in MRI im-

[3]The server is supplied by Centre Régional Informatique et d'Applications Numériques de Normandie (`https://www.criann.fr`)

ages ((img − μ)/σ) and ground truth volumes (gt/ max(gt)). The evaluation metrics are mean absolute error (MAE) and percentage MAE (PMAE). We also conducted plenty of additional experiments including cardiac data scale, cardiac slice selection, the influence of data augmentation, the hyper parameters selection etc., for more details please refer to Appendix B.

5.4.2 Results

Prediction error on 2D regression VGG16 vs. 3D regression VGG16

In the previous section (Section 5.3.2), we theoretically compare the difference of 2D convolution and 3D convolution. In this experiment, we train the 3D cardiac MRI data using 2D regression VGG and 3D regression VGG, respectively. For the sake of fairness, both these two models are trained from scratch. In Table 5.3, we compares the prediction error on test set between the 2D and 3D regression VGG16. One can find that the 2D regression VGG has smaller prediction error than the 3D ones. Thus, through this results combined with the theoretical analysis based on the previous 2D regression VGG, including training parameters, memory, and training time, we conclude that the 2D regression VGG on multi-slice data offers the best compromise. Therefore, in the following experiments, we use all 2D regression CNNs to train and predict the data in order to save time and memory.

Table 5.3 – The prediction error of cardiac structures volume: RV, MYO, LV on 2D regression VGG16 vs. 3D regression VGG16. ± is stand deviation.

Structure	**RV**		**MYO**		**LV**	
Model	**MAE(ml)** ↓	**PMAE(%)** ↓	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**
RegVGG_2D	**46.32**±42.79	**54.49**±77.70	**36.82**±38.82	**29.90**±29.76	**30.02**±33.56	**33.80**±47.66
RegVGG_3D	56.01±40.56	62.95±74.23	40.29±37.45	34.38±34.93	65.15±42.34	86.24±109.43

* The results of regression 2D VGG and 3D VGG are from one fold.
Because of GPU time limit (48 hours).

Prediction results of cardiac structure volumes

The following experiments are based on the results of 2D regression CNNs. We use regression VGG16, regression ResNetV2 and regression EfficientNet to separately train the multi-slice data to predict the volumes of three structures of the cardiac simultaneously. In the supervised learning mode, the input ground truth is the volumes of each cardiac structure. We trained the cardiac data of end diastolic (ED)

and end systolic (ES) phase together, in other word, the ED and ES of the one patient are in the same fold (training, validation and test). In Table 5.4, what they have in common is that these CNN backbones are pre-trained on the ImageNet public dataset [Deng et al., 2009]. From this table one can find that the prediction errors (MAE) are large, and the RV structure is the most difficult to predict. The regression ResNet with MAE loss has lower prediction error than the other models. So we take this model as a analysis example in the following sections.

Table 5.4 – Prediction error on volume of 3 cardiac structures using regression CNN models with 3 different loss functions, MAE loss, MSE loss and Huber loss (HL). The 3 CNN backbones are VGG16, ResNetV2, EfficientNetb2 (efn). The models are trained on 2900 training images. The results are average results of 5-fold cross validation.

Model	**MAE_RV(ml)**	**PMAE(%)**	**MAE_MYO(ml)**	**PMAE(%)**	**MAE_LV(ml)**	**PMAE(%)**
MAE loss						
Reg_VGG	50.51±39.81	65.64±92.96	41.58±34.38	36.43±37.59	35.29±29.49	40.20±52.63
Reg_ResNet	43.11±36.57	51.63±69.46	36.98±29.25	31.96±29.44	33.19±26.48	39.09±47.45
Reg_efn	49.55±40.88	60.76±83.83	36.70±32.26	33.50±36.88	33.51±26.93	39.78±52.15
MSE loss						
Reg_VGG	50.65±40.57	67.29±94.96	42.40±36.86	38.05±41.40	36.02±28.39	42.90±57.95
Reg_ResNet	43.82±34.47	56.33±79.52	38.11±29.74	34.63±33.99	33.28±26.27	42.01±53.51
Reg_efn	49.24±40.20	64.68±103.61	36.10±31.35	31.88±34.57	33.03±27.80	38.31±49.99
HL loss						
Reg_VGG	49.02±38.38	62.88±86.09	40.14±35.26	35.07±37.47	33.11±27.63	37.30±52.79
Reg_ResNet	49.29±38.12	66.20±93.30	35.13±31.27	30.99±32.11	32.75±28.30	41.57±56.69
Reg_efn	49.11±39.99	65.08±103.74	36.79±31.14	32.62±34.80	33.28±28.10	37.29±46.31

Comparison with state-of-the-art

We compared our method with the state-of-the-art (SotA) on the same ACDC dataset. In the Table 5.5, many of their results have separate EDV (left) and ESV (right). The SotA is based on direct prediction (segmentation-free) methods and segmentation methods. One can find that out method has a large gap with the already existing methods, especially segmentation ones, which is a little bit disappointing and shows that at this point, direct estimation of cardiac structures volum with vanilla CNN is a bit early. However, the comparison should be handled with care. In this table, the segmentation-based or segmentation-free methods predict the area of the cardiac structures slice by slice and then accumulate them to obtain the volume of the cardiac structures. Another point is that the experimental protocol and the test set are not the same. Therefore, there may be some bias to compare those results with the our method.

Table 5.5 – Comparison with state-of-the-art methods on ACDC dataset. The SotA methods have the separate ED (left) and ES (right) results on cardiac structures: RV, MYO, LV.

Structure	**RV**			**MYO**		**LV**		
Methods	**MAE(ml)**		**PMAE(%)**	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**		**PMAE(%)**
Segmentation-free methods								
[Luo et al., 2020a]	N/A	N/A	N/A	N/A	N/A	5.1±3.2		N/A
[Luo et al., 2020b]	8.1±3.5	4.9±3.1	N/A	N/A	N/A	9.2±4.5	5.9±4.5	N/A
[Zhen et al., 2016a]	12.4±5.2	10.9±9.7	N/A	N/A	N/A	14.8±8.9	10.2±7.9	N/A
Our method	43.11±36.57		51.63±69.46	36.98±29.25	31.96±293.44	33.19±26.48		39.09±47.45
Segmentation-based methods								
[Zheng et al., 2018]	N/A	N/A	N/A	N/A	N/A	12±9.1		N/A
[Vigneault et al., 2018]	N/A	N/A	N/A	N/A	N/A	11.1±5.3		N/A
[Liao et al., 2017]	N/A	N/A	N/A	N/A	N/A	15.8±9.6	9.9±9.5	N/A
[Ngo et al., 2017]	N/A	N/A	N/A	N/A	N/A	17.1±11.5	16.8±12.5	N/A
[Avendi et al., 2017]	16.1±13.1	14.1±16.6	N/A	N/A	N/A	17.5±16.9	19.2±20.3	N/A
[Bernard et al., 2018] *	10.6	N/A	N/A	7.1	N/A	10.4	N/A	N/A
[Isensee et al., 2017]	7.9	N/A	N/A	7.3	N/A	5.1	N/A	N/A

* This is the average segmentation-based results of all deep learning methods in ACDC challenge.

5.4.3 Discussions

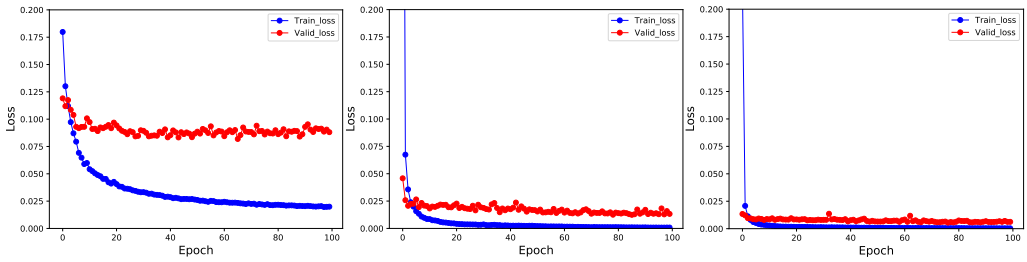
The aim of our work is to directly predict the volume of cardiac structures without intermediate segmentation steps. Based on our previous experience for predicting the fetus head circumference from 2D ultrasound images, we use the same model framework, that is regression CNNs, to implement our idea. However, in this study, we met several problems during the experiments, which we discuss below.

2D vs. 3D regression CNN model

On a theoretical level, both 2D and 3D regression CNN model can train and inference from multi-slice MRI cardiac data. The difference is that the convolution is done in a different way at the cost of a different order of magnitude of the number of parameters. On a experimental level, the 3D regression CNN model takes much more computation memory and training time than 2D regression CNN model during training on GPU server, while the model does not predict well, which indicates that the 2D regression CNN model is more practical.

Loss functions

The prediction error of regression CNN model with three regression loss functions respectively doesn't have significant difference or pattern (See Table 5.4). But during



(a) MAE (b) MSE (c) Huber Loss

Figure 5.6 – Learning curves of regression ResNet model with different loss functions. Blue curve is training loss, red curve is valid loss.

training process, the loss evolution (See Figure 5.6) witnessed that the Huber Loss converge consistently between training and validation stage. Another problem that one can observe that the model is over fitting, because the training loss is decreasing while the valid loss almost doesn't change.

Prediction results analysis

The quality of the data is uneven, which may hinder the prediction of the model. Figure 5.7 shows cardiac images of two patients. One can find that signal and noise are co-existed in each slice. The feature of three cardiac structures are not obvious and the marginal areas of the cardiac are either highlight white interference caused by the device acquiring the image, or fluid produced by the cardiac itself. From cardiac images, one can also find out that the right ventricle has irregular shape that may result in biased prediction results (Figure 5.7). The positions of the cardiac structures are also moving and thus inconsistent in these two patients.

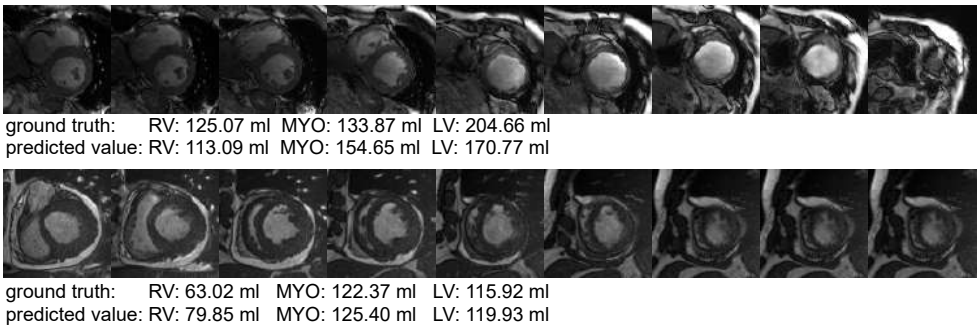


Figure 5.7 – Two cardiac data outliers predicted by regression ResNet, especially the RV structure has large bias compared to ground truth. The data are P047fm01, P053fm12 respectively.

We performed a statistical analysis of the prediction results for a test set with 50 cardiac data. Among three cardiac structures (RV, MYO, LV), RV has the highest mean absolute error predicted by regression CNN model (Figure 5.8). This is expected as it is known in the clinic to be the most difficult to estimate. The Bland-Altman plot (Figure 5.10) of predicted cardiac structure volumes also demonstrates that the large bias of three cardiac structures compared to ground truth values, especially in RV.

From a pathological point of view, in order to ensure the generalization of the model, the model is trained with the data evenly distributed according to the pathology. Based on this fact, the prediction of RV volume is also difficult in data with abnormal RV cardiac disease (Figure 5.8). Besides RV disease, the prediction bias is also large to the patients with dilated cardiomyopathy (DCM) and hypertrophic cardiomyopathy (HCM) (Figure 5.9).

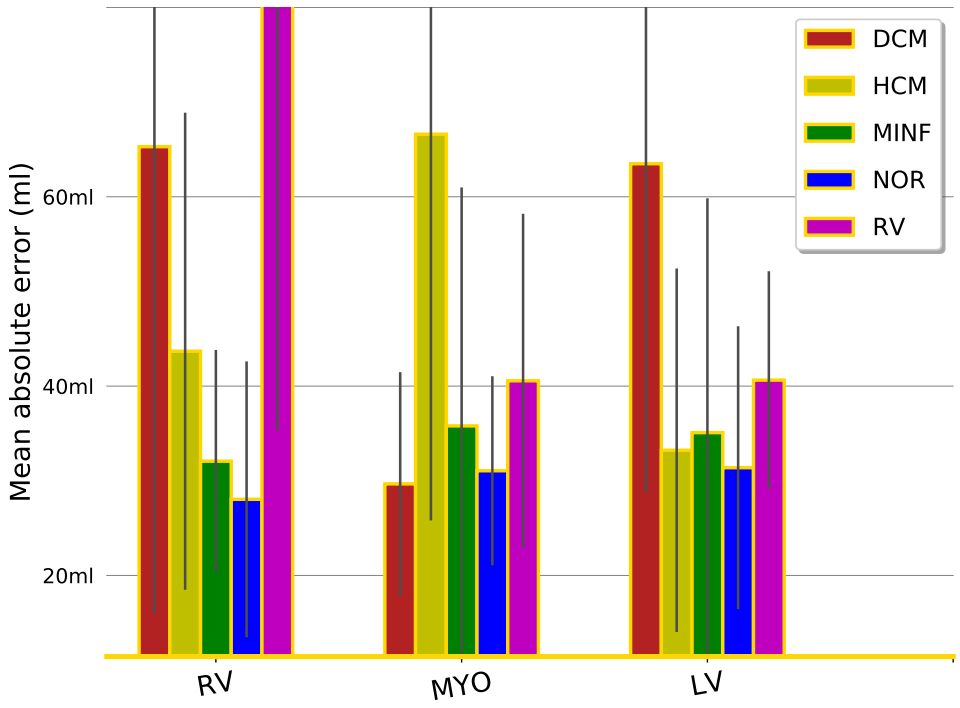


Figure 5.8 – The mean absolute error of three cardiac structures (RV, MYO, LV) according to different pathologies. The prediction results are from Regression ResNet.

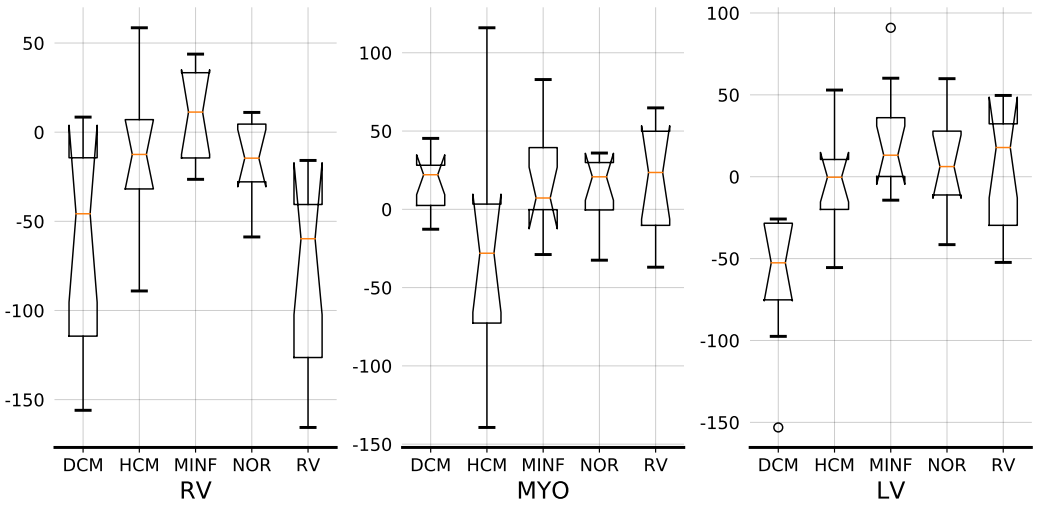


Figure 5.9 – Box plot of predicted cardiac volumes of three cardiac structures (RV, MYO, LV) according to different pathologies. The prediction results are from Regression ResNet.

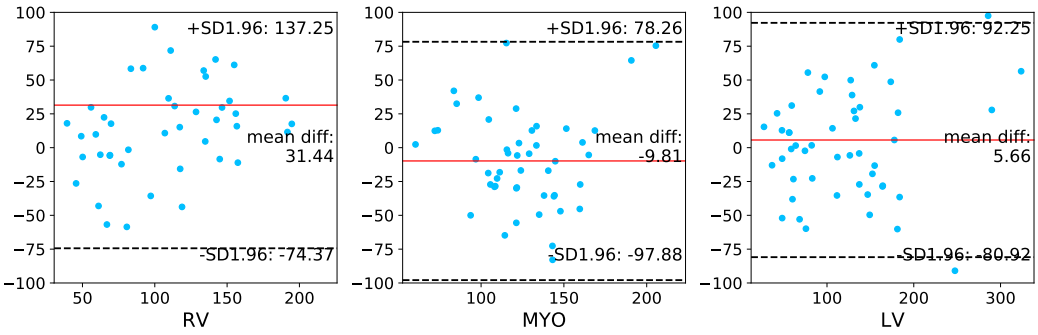
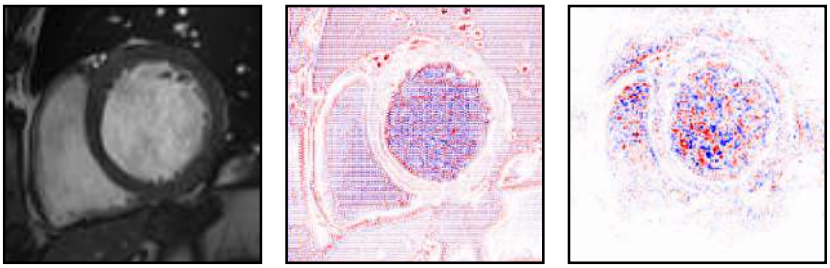


Figure 5.10 – Bland-Altman plot of cardiac structure volume predicted by proposed regression ResNet model. The x-axis represents the average value of ground truth and predicted volume; the y-axis, the difference between ground truth and predicted volume (in ml). The horizontal black dotted lines represent the upper and lower limits of 95% consistency. The middle solid red line represents the mean of the difference.

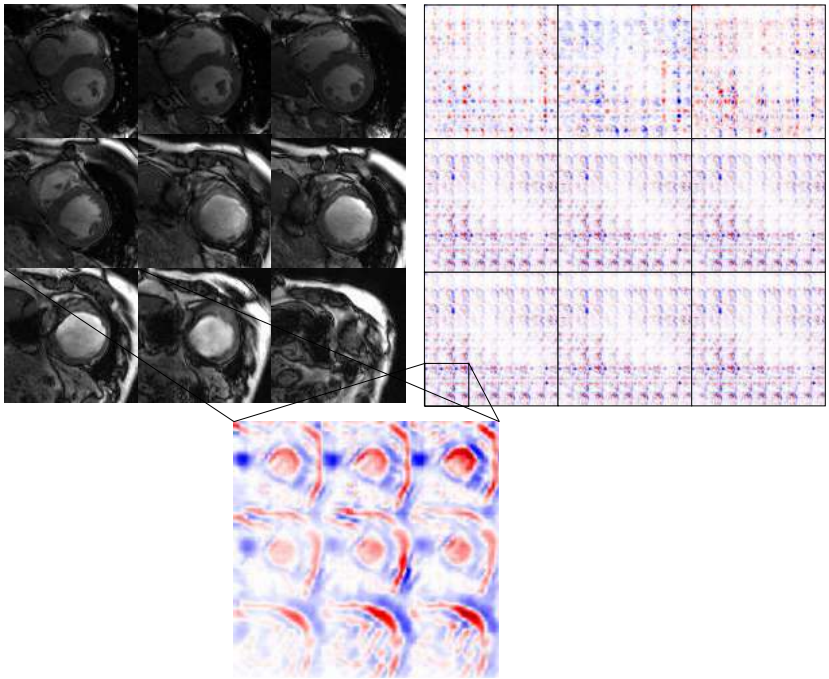
Saliency maps of Regression CNN

We generated and analyzed saliency maps of the regression CNN model for cardiac structure volume prediction using the LRP algorithm based on iNNvestigate library [Alber et al., 2019], see Figure 5.11. From the saliency maps on one slice, we can see that the model focuses on the RV and LV, while the MYO has a lesser influence in this image. Besides, the two CNN backbones EfficientNetb2 and VGG16 have different feature distributions, which reflects their ability of feature extraction. Another

groups of saliency maps are the model whose input is 9-slice data, which means the model has 9 channels at the first input layer. Therefore the subsequent convolution layers are also increased in multiples of 9. For instance, zooming in on the slice of the seventh, the white noise at the edges in cardiac image induces feature extraction from the model, and the LV structure can still be faintly seen (red area in the center), which leads to a large bias in the prediction results of the model. This demonstrates the need for noise reduction in image data, if accurate predictions are to be obtained.



(a) Single slice (P001fm01), SA_reg_efn (left), SA_reg_VGG (right)



(b) 9-slice (P047fm01), SA_reg_ResNet

Figure 5.11 – Saliency maps of regression CNN models on cardiac images. The first image is one single input cardiac slice and its saliency map (SA) of regression EfficientNet (reg_efn) and regression VGG (reg_VGG). The second image is the 9-slice input cardiac and their saliency maps of the regression ResNet model. Red color means positive contributions, blue color means negative contributions.

5.5 Conclusions

In the general practice, the ventricle cavities and the myocardium are first segmented, and then the areas and volumes are calculated based on the result of the segmentation. In this study, we investigated how regression-based CNN models can directly predict the volume of cardiac structures (RV, MYO, LV) without segmentation intermediate steps. Our method was validated on the ACDC dataset. We first preprocessed the ACDC data set by cropping and unifying the slice number to 9. Data augmentation is used, based on grid search method, to increase the amount of data. Transfer learning is applied in this study: the CNN backbones are pretrained on ImageNet. The predicted results of cardiac structures were analyzed and discussed that large bias exists, especially in RV structure. We also analysed the model's interpretability through a post-hoc explaining method. The saliency maps tells that this regression-based methods are reliable to some extent. Although the idea of direct estimation has big potential in a clinical setting, results are not fully convincing yet and the prediction error need to be further reduced. We could first explore the estimation of the area from single slice, then computing the volume. We also believe that with more adhoc or specific architectures, that would be better adapted to the data (3D, noise), could allow to enhance the results.

Chapter 6

Conclusions and future work

Contents

6.1 Conclusion

In this thesis, we have proposed some contributions in medical image segmentation, biomarker estimation through regression CNN and explainability in regression CNN, that we summarize below.

6.1.1 A loss function based on the Kappa index

The class imbalance problem cannot be ignored in image segmentation when using supervised deep learning techniques. One loss function that is already well known for solving the class imbalance is Dice loss [Milletari et al., 2016], which is based on Dice index. The Dice loss calculates the overlap area between predicted positive area and ground truth positive area. That is to say, the Dice loss does not take the background pixels into account.
We proposed a loss function which is based on Kappa index, called Kappa loss. Different from Dice loss, we consider all the pixels including the background information (negative area in prediction and ground truth). The skin lesion segmentation experiments results showed that our proposed Kappa loss can not only surpass the Dice loss by a small margin but also the model has better convergence than the U-Net with Dice loss. At this point, we have added a new member to the family of loss functions, namely the Kappa loss function.

6.1.2 Biomarker prediction

Fetus head circumference prediction

The fetus head circumference (HC) is one of key biomarkers for monitoring a fetus growing stage. Conventional fetus head circumference prediction is performed by segmentation methods. However, the segmentation-based methods require more than one step: contour segmentation and ellipse fitting, then the head circumference calculation.
In this work, a direct HC prediction approach was proposed. We utilize regression CNN model to directly predict fetus HC from ultrasound images without intermediate segmentation steps. The regression CNN is composed of a CNN backbone and a regression layer. Transfer learning strategy is used in order to improve the prediction accuracy. The loss function is regression loss (MAE loss, MSE loss or Huber loss).

Another contribution of this work is that we compared the proposed segmentation-free (regression CNNs) with segmentation-based methods in a fair experimental environment from several aspects. We used the same dataset (HC18 [van den Heuvel et al., 2018b]) including data preprocessing and dataset split and GPU server to train and estimate the HC value. We evaluated the explainability of regression CNNs, the prediction error of two approaches, the theoretical memory as well as practical computation efficiency of two models, the learning curves of two models during training, and agreement analysis of two prediction results. The experiments results of segmentation-free methods are comparable to that of segmentation–based methods although improvement room is left. Nevertheless, the HC prediction error of both segmentation-based and segmentation-free methods are smaller than the manual variability.

Cardiac structure volume prediction

There is a more complicated case about direct biomarker prediction is cardiac multi-structure volume prediction from 3D MR imaging.
With so many lives lost each year due to cardiovascular disease, a quick and effective examination of the patient's heart is critical, but the patient to doctor ratio varies from region to region and hospital to hospital. Therefore, designing automated and effective diagnostic methods to assist physicians can greatly reduce the amount of effort physicians spend on an individual patient, allowing for early prevention or treatment of the patient.
In this study, we utilized regression CNNs to directly predict the cardiac structure (RV, MYO, LV) volumes. Before training the the models on ACDC dataset [Bernard et al., 2018], we performed preprocessing the data in the aspects of cardiac area cropping, slice number uniforming as well as data augmentation given that each subject has different shape in the original dataset and the amount of the data is small. For the regression CNN model, transfer learning is also used in our method, for which the CNN backbones are pretrained on the natural images dataset (ImageNet [Deng et al., 2009]). But in this study, we adapted the proposed regression CNN models from RGB channel to multi-channel to fit in the training data in order to copy the weights from pretrained CNN backbones. Several experiments have been conducted and analyzed. The experiments have promising results except for the volume of RV structure which is difficult to estimate.

6.1.3 Explainable AI in medical imaging

Saliency maps of regression CNN models in medical images

In this thesis, we made a survey about explainable AI (XAI). In the specific application of XAI, we generated saliency maps from several post-hoc explaining methods on the regression CNNs. In the HC prediction problem, we utilized explaining methods to valid the interpretability of regression CNN models in the form of saliency maps. The experiment results indicated that the highlighted areas of saliency maps match the contours of the fetus head as observed by the human eye, and these highlighted areas are the main contribution to the predictions made by the model. Therefore, we can know that the regression CNN has the ability to learn features and to make predictions based on that feature. We also validated the explainability of regression CNNs on 3D cardiac imaging using one explaining method.

Evaluation metrics of explaining methods in regression CNNs

Besides the saliency maps that can visually show the highlighted features learned by deep learning models generated by certain explaining method. There is another method that can quantitatively evaluate each explaining method based on the perturbation method. In our study, we adapted the criteria (AOPC score) from classification CNN to regression CNN. On the one hand, we used this criteria to evaluate if an explaining method is effective. On the other hand, it can be used to evaluate of one regression CNN model is better than others.

6.2 Future work

In the future work, we will explore more possibilities in two levels:

- Technical level;
- Medical imaging level.

6.2.1 Technology innovation

Geometric deep learning

Most of current deep learning models are based on data-driven, in other words, the model's performance is excellent as long as the data has huge amount and

good quality so that the model can learn various features/information from them. Geometric deep learning is intended to avoid *The Curse of Dimensionality* [Indyk and Motwani, 1998] by the idea of symmetry prior to keep the data invariance in the forms of graphs or grids or other mathematical representations. Thus, this technology will be a study topic in the future.

Attention mechanisms in computer vision

Attention mechanisms are originated in the field of natural language processing. It was later applied to the field of computer vision and achieved remarkable results. In particular, the Transformer model [Vaswani et al., 2017] and Vision Transformer [Dosovitskiy et al., 2020] of recent years, which has only attention structures, completely replaces the convolutional neural layer. Their performances surpass the CNN based deep learning models. In addition, to some extent, the attention mechanism model also carries a self-explanatory property, which can improve the credibility of the model. Therefore, its application to medical image analysis is of great interest.

Making the deep learning models explainable

In deep learning-based medical image analysis, the prediction results made by the model are required to be as accurate and trustworthy as possible, otherwise serious medical incidents may occur. This requires us to interpret the model post-hoc on the one hand, and on the other hand to make the model capable of self-interpretation.

6.2.2 Medical imaging problems in practice

Verify the other medical datasets

In future works, we plan to verify the proposed segmentation-free (regression based) methods on the other medical datasets. Despite the success of this method in fetal head circumference prediction, we hope that it can obtain similar results to the segmentation-based method on more other organs or tissues of medical images. At that time, new problems and challenges may be encountered, such as the pre-processing of images, or the generalization ability of the model, which we will analyze specifically based on the specific problem.

Making the methods applicable in clinical application

Because advanced technologies are created for practical problems, we im in future work, to investigate the needs of real medical problems, for example, from the problem of pre-processing medical images generated in the machine, to the physician's expectation to get specialized medical images. This will enable doctors to focus on the patient itself and alleviate the time and effort spent on the other tasks. Another perspective is to implement deep learning technologies into clinical medicine applications, designing simple, reliable and effective automated algorithms to assist doctors in diagnosing or examining patients.

Part III

Appendix

Appendix A

The explainability of regression CNNs

Contents

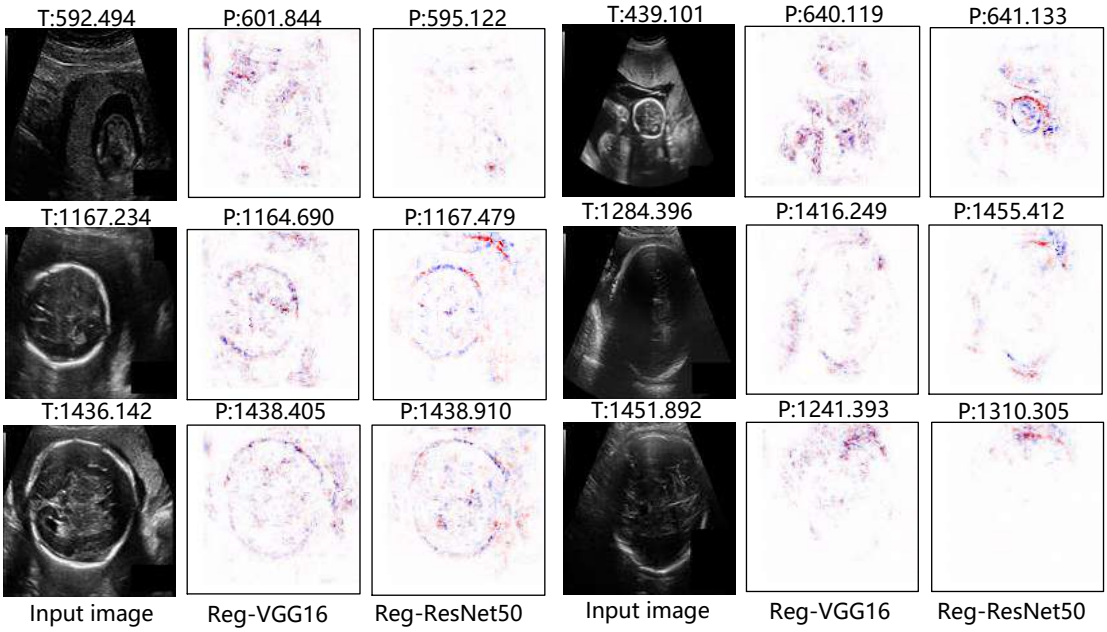
A.1 The explainability of regression models

A.1.1 The explainability of regression VGG and regression ResNet

Last section describes the performance of each explanation methods. Now, we can utilize these explanation methods to compare different regression CNN models.

As shown in Figure 4.6, both regression VGG16 and regression ResNet50 are successful in learning the features from ultrasound images to assess the HC. From Table 4.4, we can gather that the regression ResNet50 has slight better performance on the whole, since AOPC values are larger in absolute value.

A.1.2 Saliency maps for correct vs incorrect prediction



(a) Good prediction (b) Bad prediction

Figure A.1 – Saliency map of Reg-VGG16 and Reg-ResNet50 with Input*Gradient explanation method. P and T: resp. predicted and ground truth HC values (pixels).

In this experiment, we arbitrarily pick one of the best performing methods from the previous results, and thus the use Input*Gradient explanation method to generate saliency maps from images with small prediction error (Figure A.1 (a)), and with large prediction error (Figure A.1 (b)). We can see that the well predicted images have obvious head contour, at least in the 2 last rows of Figure A.1 (a). The models are able to learn the features from these images, therefore the saliency maps show

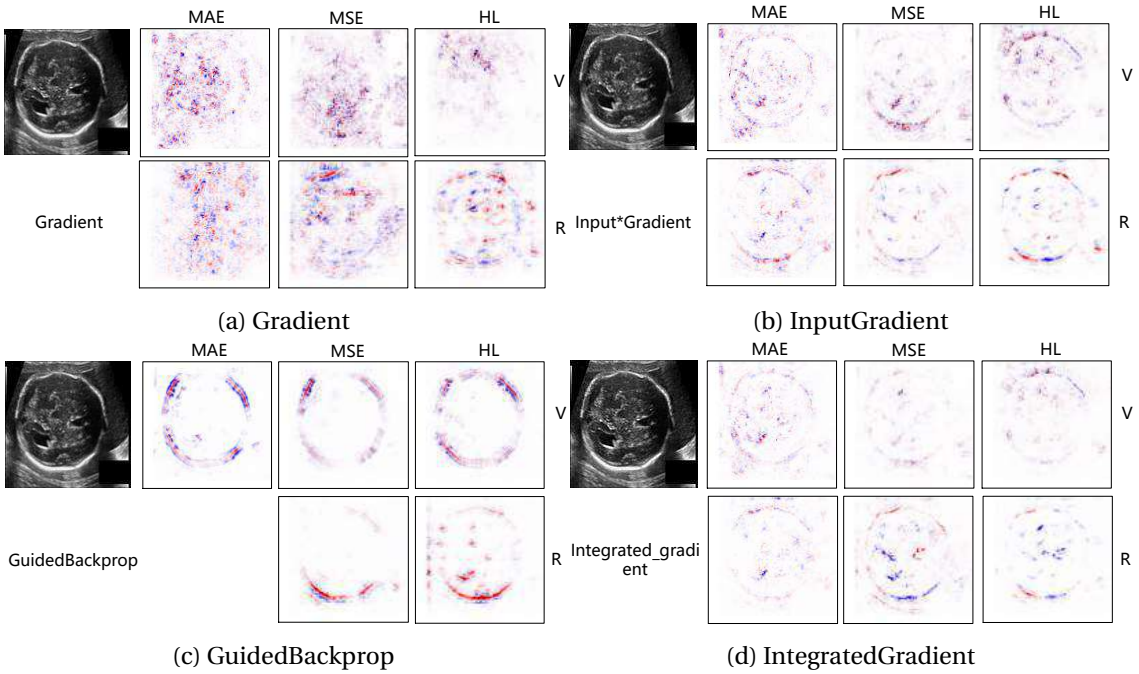
key features. However, it is not always the case: the first row shows a small prediction error, and the head contour are not specifically highlighted. For the badly predicted images, the saliency maps highlight features that are spread and not localized into meaningful segments. The models can not learn the features from these images. However, beyond the score, it seems to be related to the quality of the images: This probably due to irregular and blurry head features, and fan-shaped areas existed in the images which could affect the decision of model. Therefore, performing image preprocessing before training is an effective way to improve the performance of models.

A.1.3 Comparison of saliency maps for different loss functions

In addition to comparing the saliency maps of different regression CNN models as well as the saliency maps on good/bad prediction results. We further compare the performance of different regression loss functions in regression CNNs through different saliency maps. We use 8 different explanation methods to generate saliency maps on regression VGG16 and regression ResNet50 tested on one same input US fetus head image with MAE loss, MSE loss, and Huber loss, respectively. See Figure A.2 and Figure A.3.

Through those two figures, several finding can be concluded:

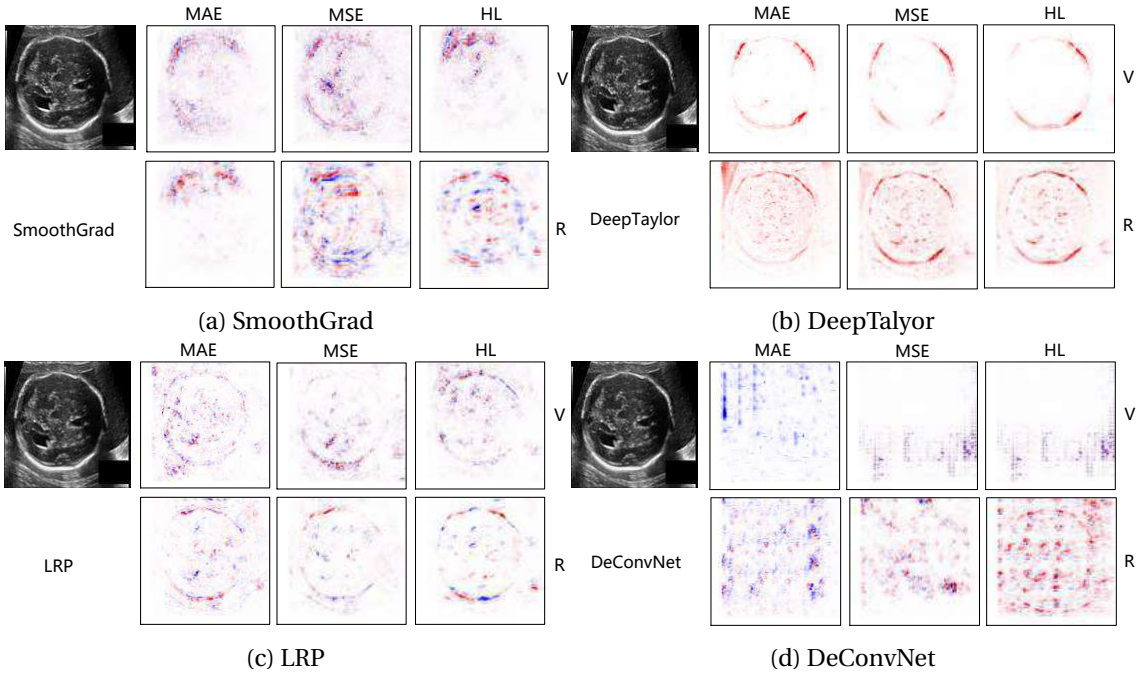
- The MSE loss is slightly sensitive than the MAE loss (See Figure A.2(d), Figure A.3(a),(b)). This is due to the square item in the MSE loss, which will change obviously than MAE loss with absolute item.
- Because the Huber loss is a compound loss of MAE loss and MSE loss with a weight value between them. It performs alike with MAE loss and MSE loss.
- As has been discussed before, the regression CNN models and explanation methods have different performance in each saliency maps, which can help to select the better ones. This groups of figures follow the same rule.



(a) Gradient (b) InputGradient

(c) GuidedBackprop (d) IntegratedGradient

Figure A.2 – Saliency maps of different explanation methods under 3 different loss functions and regression CNN model VGG16 (V) and ResNet50 (R).



(a) SmoothGrad (b) DeepTalyor

(c) LRP (d) DeConvNet

Figure A.3 – Saliency maps of different explanation methods under 3 different loss functions and regression CNN model VGG16 (V) and ResNet50 (R)

A.1.4 Comparison of AOPC scores for different loss functions

The other supplementary experiment including comparing AOPC scores of two regression models with 3 loss functions and prediction error maps of different analysis methods by adding perturbations. See Figure A.4 and Table A.1.

Quantitative analysis

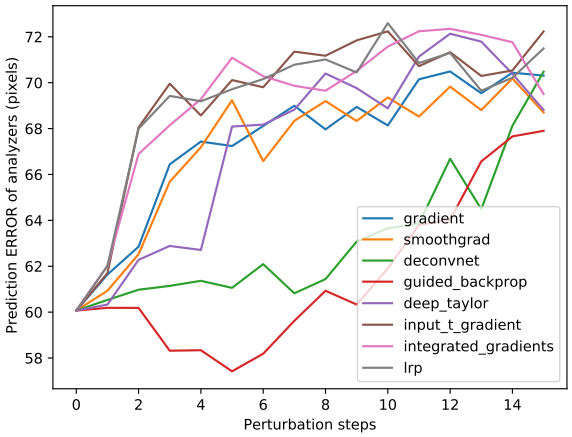
Table A.1 is the experiment of adding perturbation on test images (200 images), then the regression CNN models go through each explanation method. For instance, in Gradient explanation method (The second column), the Regression ResNet with Huber loss has the lowest AOPC value (-24.17), which means this model is the most sensitive than the others. That is to say, the Regression ResNet with Huber loss identify the right feature from test images. Therefore, one can known which loss function is more suitable in this way.

Table A.1 – Performance (AOPC scores) of different explanation methods after perturbation, with two regression models and three loss functions. G: Gradient, SG: SmoothGrad, DCN: DeConvNet, DT: DeepTaylor, GB: GuidedBackprop, I*G: Input*Gradient, IG: IntegratedGradients. Lower is better. Best scores in bold.

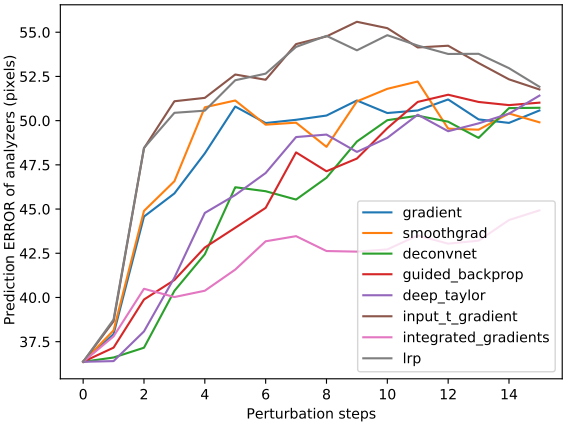
Model	**G**	**SG**	**DCN**	**DT**	**GB**	**I*G**	**IG**	**LRP**
RegVGG_MAE	-7.31	-7.39	-2.87	-7.40	-1.66	**-9.19**	**-9.49**	**-9.17**
RegVGG_MSE	-7.80	-7.18	-5.36	-9.10	-2.99	**-14.57**	N/A	**-14.46**
RegVGG_HL	-23.39	-21.63	-24.59	-27.78	-18.86	**-29.47**	N/A	**-29.27**
RegResNet_MAE	-11.53	-11.84	-9.25	-9.89	-9.72	**-14.75**	-5.60	**-14.58**
RegResNet_MSE	-11.31	N/A	-11.18	-19.41	N/A	**-32.48**	**-20.49**	**-32.51**
RegResNet_HL	-24.17	-24.27	N/A	-22.66	-28.42	**-37.12**	-22.81	**-38.12**

Qualitative analysis

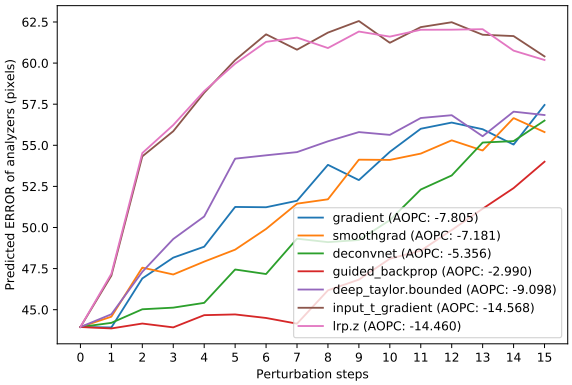
Figure A.4 shows the perturbation process on two different regression CNN models with three different loss functions respectively. The test images are divided into 16 subareas, the perturbation are added on each subarea one by one based on the importance of prediction score. One can find the prediction error becomes higher after the feature is blocked by the perturbation in most of curves. The steepest curve demonstrates that this explanation method is the most effective to capture the feature of images that the regression CNN has learned.



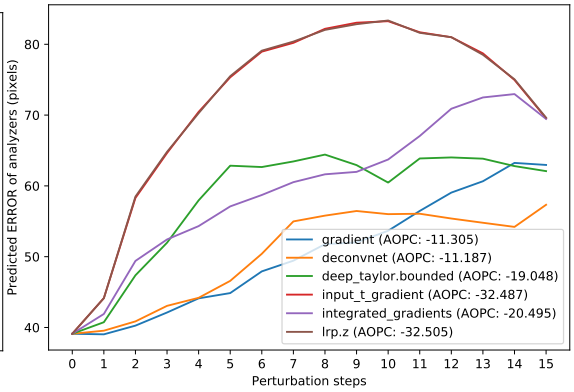
(a) VGG_MAE



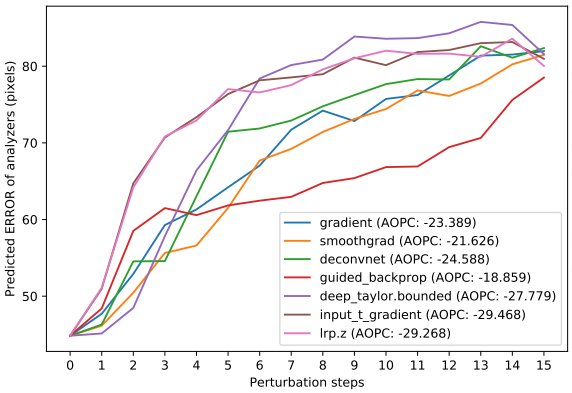
(b) ResNet_MAE



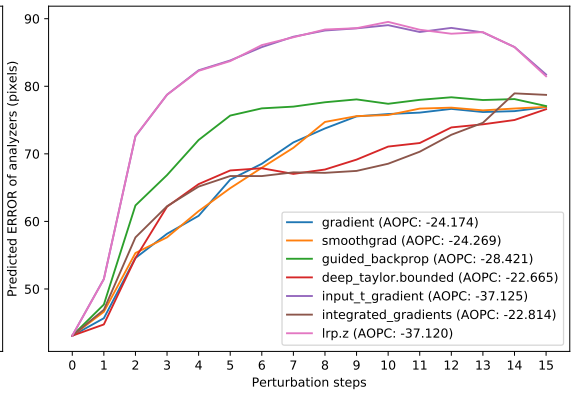
(c) VGG_MSE



(d) ResNet_MSE



(e) VGG_HL



(f) ResNet_HL

Figure A.4 – Perturbation steps of different analyzers under Regression VGG16 and Regression ResNet model with loss function MAE, MSE, HL, repectively.

A.2 Conclusion

Understanding whether the model can learn the relevant features in images and take the right decision is crucial in the medical domain. Whereas there have been a wealth of works in classification networks, there is a void for interpreting regression networks.

In this study, we address the problem of estimating the head circumference in fetal head directly from US images. We use several post-hoc explanation techniques that produce saliency maps and adapt a perturbation based quantitative evaluation method, to assess the relevance of the saliency maps. We also investigate the explainability of regression losses including the MAE loss, MSE loss and Huber loss.

The experimental results proved that the regression CNN models are able to learn the key features from the input ultrasound fetus images, and in particular, the head circumference. One finding is that for this application, Gradient and DeConvNet method are particularly insensitive to different CNN models or data, and that ResNet50 seem to have better learnt the head features. Thus so far, we have extended the model property from classification to regression and explored a specific regression task.

Moreover, we should not only explain the model but also get some feedback according to explanation results, for example, in our case, the content of images can also affect the model's decision, because for those images that the model have bad predictions, the explanation methods cannot show clear features, neither. Finally, the performance of explanation methods used in this work are different from each to others. However, relying only on the saliency maps or on the perturbation methods is far from being enough to get insights from a black box. Despite explanation methods have emerged, there is room for improvement as they are not yet mature enough but this is a step toward more reliable and safe deep learning approaches in medical fields.

Appendix B

Additional experiments on ACDC dataset

Contents

B.1 The influence of data modality

The data modality in this study refers to the different slices of cardiac data, the different training scale. In this section, we explore the influence of different data modalities. Besides, we respectively valid the prediction ability of regression CNN on single cardiac structure and multi cardiac structures.

B.1.1 Selection of cardiac slices

Motivation

In short axis view of cardiac MR images, each slice represents a part of cardiac. All the slices stacked up is a complete cardiac. However, when using deep learning models to learn relevant features from these slices, it is not necessarily the case that the more complete the information is, the better; instead, some redundant slices can lead to prediction errors caused by noise. For example, in the ACDC dataset, some of the slices of the subject contain little or no information about the cardiac structure. For this reason, feature extraction may benefit from appropriate streamlining of the cardiac slices. In related works, [Luo et al., 2017] explored various combinations of slices from single image at different position to two images, and then three images etc. Their experiments results showed that when the input view is the combination of Top+Middle+Bottom slice from a cardiac, the model has the best performance.

Experiments on different slice combinations

In this experiment, we explored different slice combinations on ACDC dataset inspired by [Luo et al., 2017]. We respectively take the Top slice, Top+Middle+Bottom slices (the 2^{th}, 3^{th}, 4^{th} slice of cardiac), Top three slices (2^{th}, 3^{th}, 4^{th}), Middle three slices (4^{th}, 5^{th}, 6^{th}), Bottom three slices (6^{th}, 7^{th}, 8^{th}), and the entire 9 slices. Experimental results (Table B.1) showed that the model's performance is better when the input training data has three slices of a cardiac, which implies that too few slice (one slice only) or too many slices (9 slices) does not bring effective information to the regression CNN models.

Table B.1 – Prediction error (mean absolute error, MAE) on volume of 3 cardiac structures from different slice combinations using regression VGG model. The numbers in brackets are the serial numbers of the slices.

Input views	**MAE_RV(ml)**	**MAE_MYO(ml)**	**MAE_LV(ml)**
Single Slice (Top)	46.81	37.06	50.24
3-slice $(258)^{th}$	36.53	31.33	28.45
3-slice $(234)^{th}$	31.44	27.58	28.31
3-slice $(456)^{th}$	37.77	25.03	23.40
3-slice $(678)^{th}$	36.55	27.97	26.03
Entire slices $(1\text{-}9)^{th}$	46.83	39.80	36.43

B.1.2 Different training data scale

Since medical images are more complex to acquire and pre-process than natural images, and the annotation of targets in medical images is labouring work. The insufficient amount of data is a big obstacle for data-driven deep neural network based models. Data augmentation can remedy the problem of insufficient original data. We use grid search method (Algorithm 5.3) to generate different images.

We use regression VGG16 to predict the volumes of three structures of the cardiac from 1000, 2000 and 2900 training images respectively. From Table B.2, the model's prediction error decreased clearly on 2000 training images than that on 1000 images. However, when the training images are 2900, the model's prediction error get larger, which indicated that not the more training data, the model has better performance.

Table B.2 – Prediction error on volume of 3 cardiac structures with different input data scale using regression VGG16. ± is stand deviation. The models are trained on 1000, 2000, and 2900 training images, separately.

# of images	**MAE_RV(ml)**	**PMAE(%)**	**MAE_MYO(ml)**	**PMAE(%)**	**MAE_LV(ml)**	**PMAE(%)**
1000	47.45±38.33	60.29±83.00	39.67±33.66	34.09±33.55	35.88±28.35	44.62±57.82
2000	44.75±39.20	54.62±72.54	39.39±34.13	33.88±34.63	32.71±27.14	37.77±46.85
2900	50.51±39.81	65.64±92.96	41.58±34.38	36.43±37.59	35.29±29.49	40.20±52.63

In this experiment, we evaluate the prediction error of regression VGG that is trained on different number of augmented training data. In Figure B.1, we can see that when there is only original training data, the prediction error is pretty high. As the augmented data gradually increased, the prediction error of the model decreased until it reached a very small value, and then the error began to increase again. This indicates that on the one hand, training a CNN model requires sufficient

amount of data, and on the other hand, it also points out that negative effects may occur when there is too much homogeneous augmented data. Same observation results and conclusion is given in [Huang et al., 2021a].

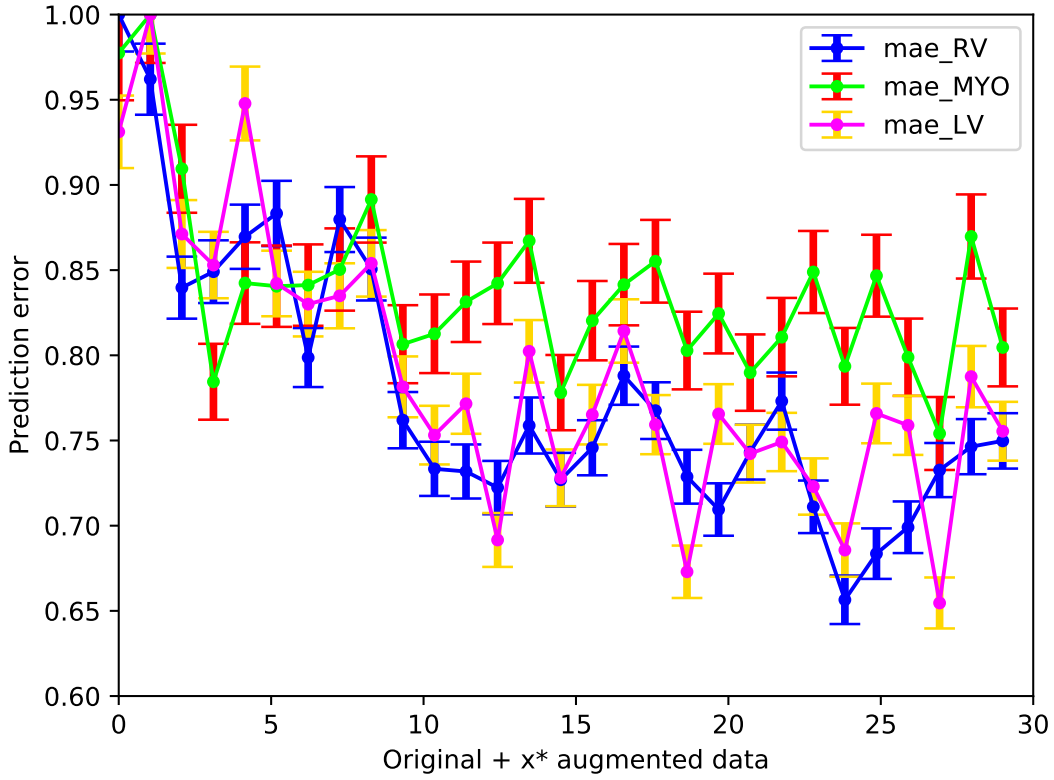


Figure B.1 – Prediction error with the influence of different number of augmented training data. The horizontal axis stands for the amount of original training data plus x times of augmented data. The vertical axis is the prediction error rate.

B.1.3 Single cardiac structure prediction vs. Multi-structure

In the above experiments, the three structures of cardiac are estimated simultaneously by regression CNNs. In this experiment, we explore the performance on single cardiac structure prediction. See the table below (Table B.3). One can find that the model's prediction error on single structure is lower than the multi-structure prediction. That makes sense because predicting one target once is easier than predicting multi-targets at the same time for a regression CNN model. Despite that, in the cardiac multi-structure estimation case, it's more practical for applying multi-structure prediction model.

Table B.3 – The prediction error of each single cardiac structure and multi-structure on regression VGG16.

	RV		MYO		LV	
	MAE(ml)	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**
Multi	50.51±39.81	65.64±92.96	41.58±34.38	36.43±37.59	35.29±29.49	40.20±52.63
Single	**44.25**±39.21	**47.87**±55.91	**41.16**±34.99	**33.34**±30.36	**29.95**±25.60	**34.38**±42.02

B.2 Determination of hyper parameters

In this section, we explore the hyper parameters in three aspects. First, in model training level, we valid different batchsize and learning rate; Second, in training data level, we valid different ratio of training, validation and test set; Third, we study the influence of data type (pathologies in ACDC dataset) distributed in each training, validation and test set.

B.2.1 Batchsize and learning rate

When training a dataset, one need to specify how many epochs this model is to be trained on the training set. In general, the epoch is chosen to be between 100 and 200, depending on experience. If it is too small, the model is not trained sufficiently, and if it is too large, it will take too long and the loss will no longer decrease. In practical situations, when the computing power of the device is average, the model cannot train all the data at once in one epoch. Therefore, the dataset can be divided into small batches to be trained one by one according to the computing power, the batchsize is the number of images in one batch.

In deep learning techniques, updating the weights of each neuron of the neural networks is achieved by means of a specified optimization algorithm, such as Adam optimizer [Kingma and Ba, 2014]. The weight update also has a rate, i.e., a learning rate. The learning rate is as important as the optimization algorithm. If it is too large, the optimization will diverge; if it is too small, the training will take too long or we will end up with sub-optimal results.

In this experiment, we explored the different learning rates and batchsize (Table B.4). When the learning rate is set large, the prediction error on volume of cardiac structures is also larger than the other two groups. One reason could be the model cannot optimize well if the model learns too rush. Another reason could be the model is pretrained on ImageNet, in that case the model is trained with the learning

rate 1e-4, then the pretrained weights in the model become confused with the larger learning rate during training. For batch size, frankly, no clear pattern can be found from this table. It depends on the learning rate actually.

Table B.4 – Prediction error (mean absolute error in milliliter, MAE) on volume of 3 cardiac structures with different parameter settings in the aspects of batchsize and learning rate using regression VGG model. ± is standard deviation.

lr	**1e-3**			**1e-4**			**1e-5**		
bs	**RV**	**MYO**	**LV**	**RV**	**MYO**	**LV**	**RV**	**MYO**	**LV**
4	55.9±40.1	41.2±35.1	65.4±49.2	47.6±41.2	38.4±32.9	32.8±26.7	44.2±37.3	38.5±30.7	34.1±28.1
8	54.0±40.0	42.7±34.1	60.7±45.0	48.0±42.2	40.3±35.3	33.8±29.4	45.4±37.2	37.1±30.8	33.9±27.9
12	51.3±40.3	46.0±39.5	43.3±34.3	44.9±37.0	40.4±32.5	32.9±25.9	43.9±35.8	37.3±30.9	34.1±26.6
16	48.7±39.1	42.7±36.4	39.2±29.5	49.4±42.0	39.6±34.5	34.1±28.2	46.1±37.3	39.2±31.2	36.5±29.7
20	45.8±38.0	43.1±35.9	37.2±30.8	43.8±38.8	40.7±33.5	32.2±25.8	44.5±38.1	36.9±31.3	35.5±28.3
30	47.1±39.1	39.8±33.3	36.3±26.8	46.2±38.9	39.0±31.8	34.1±27.2	45.1±38.6	38.6±32.1	36.1±28.7
40	49.4±39.7	41.2±33.5	37.5±27.8	46.9±37.0	38.3±30.8	34.1±25.2	45.5±38.1	38.3±31.0	37.0±29.7

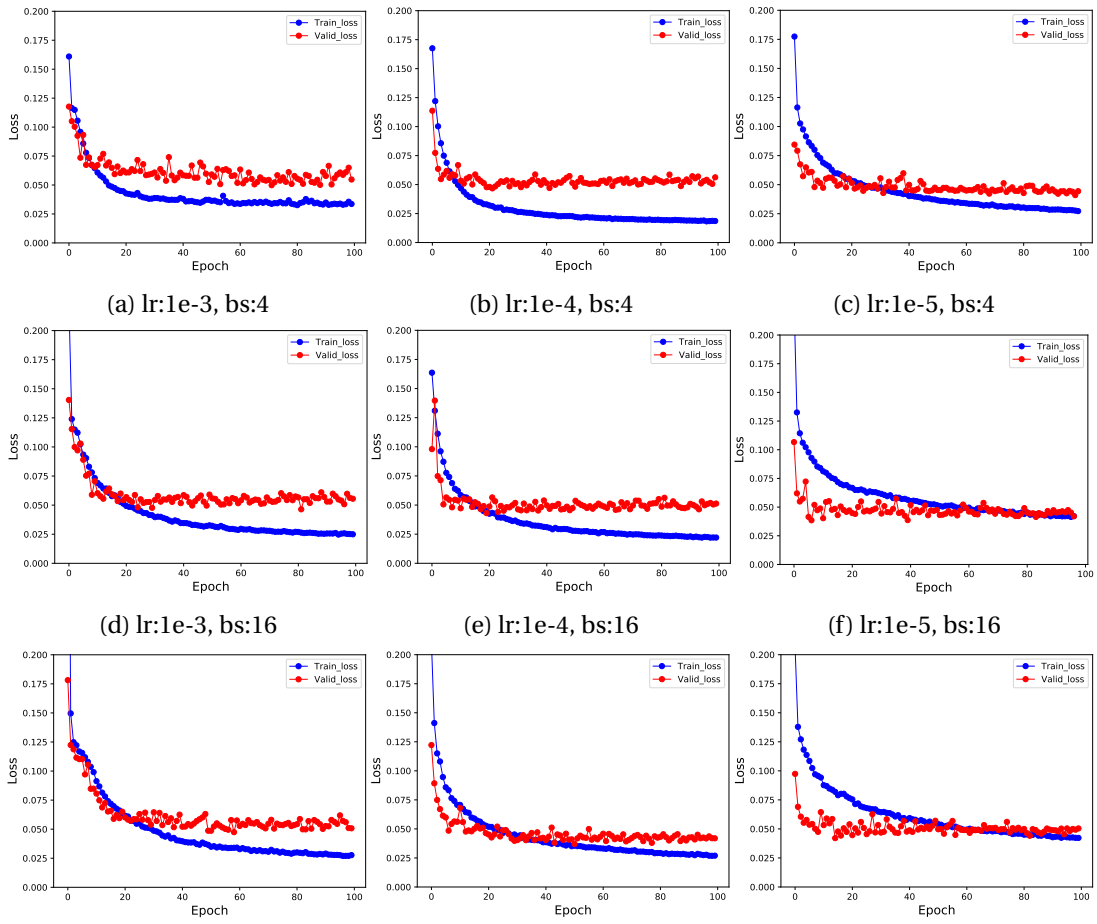
Furthermore, we describe the learning curves with respect to different learning rates and batchsize. See Figure B.2. One can find that when the learning rate is small (1e-5), the model learns slowly and doesn't converge yet within 100 epochs. For the batchsize, it seems that a larger batchsize can reduce the gap between training loss and valid loss. The reason for this phenomenon is to be demonstrated by further research.

B.2.2 Dataset splitting

In order to train any machine learning model, no matter what type of dataset is used, one must split the dataset into training data and test data, and a small part of data for validation. When splitting a dataset there are two competing concerns:

- If the training data is less, the model's performance may have greater variance. Because the model does not recognize new and unseen data very well.
- If the testing data is less, the model's performance statistic will have greater variance.

Thus, the data should be split in such a way that neither is too high, it depends more on the amount of data at hand. Because the number of medical images is limited, in addition to do cross-validation to the data, it is crucial to choose the appropriate splitting ratio.



(a) lr:1e-3, bs:4 (b) lr:1e-4, bs:4 (c) lr:1e-5, bs:4

(d) lr:1e-3, bs:16 (e) lr:1e-4, bs:16 (f) lr:1e-5, bs:16

(g) lr:1e-3, bs:40 (h) lr:1e-4, bs:40 (i) lr:1e-5, bs:40

Figure B.2 – Learning curves (loss changing) of different learning rates (ls) and batchsize (bs) during training stage. Blue dotted line is training loss, red dotted line is valid loss.

In this experiment, we test several data splitting ways. The prediction error of the model is shown in Table B.5. When the training data is sufficient, the prediction error is small, that is with data augmentation, the prediction error is decreased with larger ratio on training data, which is because the model is generalized very well through a great deal of training data. While when the training data is insufficient, that is without data augmentation, the prediction error is large and unstable, which is as result of not well generalized model and less statistics samples.

B.2.3 Data type distribution

During the experiment, we found that the RV structure is difficult to predict than the other two structures (MYO and LV). This may due to the training data and test

Table B.5 – Prediction error (mean absolute error in milliliter, MAE) on volume of 3 cardiac structures from different data splitting settings using regression VGG model with/without data augmentation. The data is split into (train, valid, test). ± is standard deviation.

	with data augmentation			**without data augmentation**		
Data split	**RV**	**MYO**	**LV**	**RV**	**MYO**	**LV**
(70,52,60)	51.9±41.8	34.9±32.4	36.4±27.3	52.6±41.4	40.6±36.2	43.9±35.7
(100,32,50)	45.0±36.0	36.0±30.8	32.6±25.1	48.8±34.4	32.4±28.1	40.0±30.7
(110,32,40)	43.1±32.0	28.3±24.9	29.2±21.5	48.4±32.8	31.9±26.1	40.3±29.9
(120,32,30)	35.7±31.5	29.0±27.6	28.2±22.6	51.6±36.5	35.5±35.0	41.9±34.0
(130,32,20)	37.5±33.1	28.8±23.0	27.2±23.0	41.6±32.9	36.9±33.5	36.0±27.2
(140,32,10)	38.3±28.2	30.8±23.7	26.9±17.5	45.1±34.2	33.5±26.3	39.1±26.4

data are not evenly distributed with respect to the each disease type. Since in ACDC dataset, there are 5 types of cardiac, i.e. 4 diseases and 1 normal type. For this, we try to take the same ratio of patients in each type in training and test set. So that the model can be trained in a relatively generic mode. Table B.6 summarize 182 subjects in ACDC dataset.

Table B.6 – Data distribution in ACDC dataset based on pathology.

Number	**Pathology**	**Train**	**Valid**	**Test**
36	DCM	20	6	10
38	HCM	20	8	10
38	MINF	20	8	10
34	NOR	20	4	10
36	RV	20	6	10
Total				
182	5	100	32	50

Table B.7 compares the prediction error between the evenly distributed training and test data and randomly distributed training and test data with respect of 5 types of pathologies. From this table one can see that the performance of regression CNN models have a little improvement when the training data have the same amount of each disease type. The experimental results demonstrate that the idea of making training and test data evenly distributed in various data types is good for model's generalization ability.

Table B.7 – The prediction error of cardiac structure on regression VGG16, ResNet50 and EfficientNet (efn) respectively using evenly distributed data and randomly distributed data with respect to 5 types of pathologies respectively.

	RV		MYO		LV	
	MAE(ml)	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**
Random distribution						
Reg_VGG	50.51±39.81	65.64±92.96	41.58±34.38	36.43±37.59	35.29±29.49	40.20±52.63
Reg_ResNet	43.11±36.57	51.63±69.46	36.98±29.25	31.96±29.44	33.19±26.48	39.09±47.45
Reg_efn	49.55±40.88	60.76±83.83	36.70±32.26	33.50±36.88	33.51±26.93	39.78±52.15
Evenly distribution						
Reg_VGG	45.0±36.0	55.6±73.8	36.0±30.8	32.0±33.4	32.6±25.1	40.6±57.6
Reg_ResNet	36.1±30.6	41.9±53.0	26.2±23.2	22.4±23.4	26.9±24.0	31.9±43.8
Reg_efn	39.7±33.2	47.0±63.8	27.4±25.5	23.6±25.0	30.4±24.0	33.6±42.3

B.3 Estimating cardiac volume using Transformer

B.3.1 Transformer

Self-Attention mechanism

The attention mechanism in deep learning can be broadly interpreted as a vector of importance weights: to predict or infer an element, such as a pixel in an image or a word in a sentence, we use an attention vector to estimate the degree of its association with other elements and use the weighted sum of their values as an approximation of the target. Attention mechanisms have evolved to the point where there are many categories [Weng, 2018]. In Transformer model [Vaswani et al., 2017], the authors use Self-Attention mechanism, whose mathematical definition is as below:

$$\text{Attention} = \text{Softmax}(\frac{\text{QK}^\text{T}}{\sqrt{d_k}})\text{V} \tag{B.1}$$

A common understanding is that the same matrix is given 3 names Q, K, V[1]. Two of the matrices (Matrix Q and Transposition of matrix K) do the dot product, then normalized (Softmax), and then multiplied with the third matrix (V). d_k is the dimension of matrix K. The dot product is divided by the scaling factor $\sqrt{d_k}$ so that the gradient value remains stable during the training process (avoid gradient vanishing). The geometric meaning of dot product is the angle between two vectors, the projection of one vector onto the other vector. A large value of the projection indi-

[1]The names of Q, K, V are based on the concept of information retrieval system, where Q means Query, K means Key, V means Value.

cates that the two vectors are highly correlated. Thus, to put it bluntly, the attention mechanism measures the similarity of two matrices.

Multi-Head Attention mechanism

In Transformer model, Multi-Head Attention is used, which is adding all the heads (h) together, each head is an Attention. In order to fit/optimize the model, trainable weights matrices are multiplied with each head as well as the whole Multihead.

$$\begin{aligned} &\text{Multihead} = \text{C}oncat(head_1, head_2, \cdots, head_h)\text{W}^o \\ &\text{where} \quad \text{head}_\text{i} = \text{A}ttention(\text{QW}_i^\text{Q}, \text{KW}_i^\text{K}, \text{VW}_i^\text{V}) \end{aligned} \tag{B.2}$$

B.3.2 Vision Transformer

The architecture of Vision Transformer

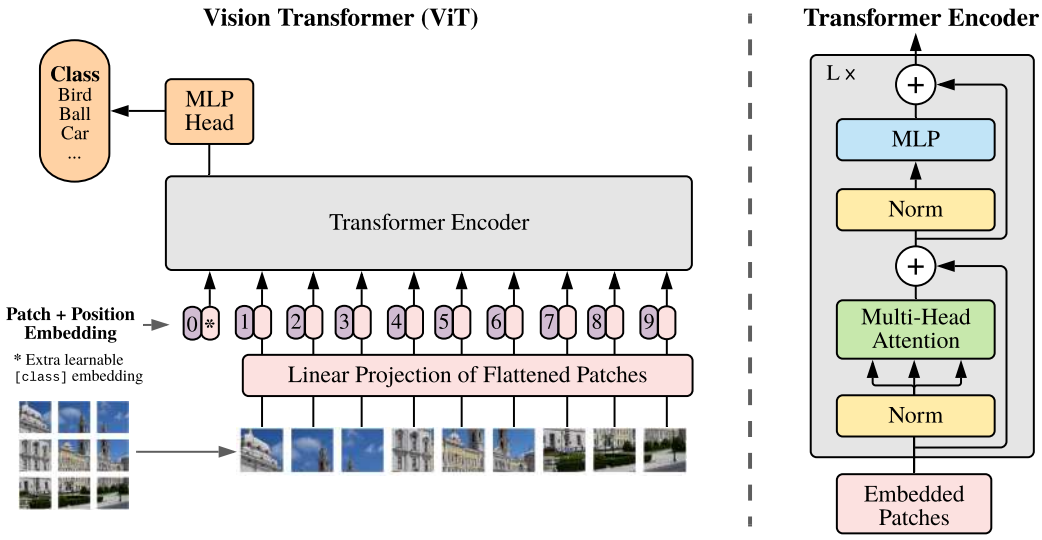


Figure B.3 – Vision Transformer model. An image is split into certain number of patches. The position information is added in each patch. A learnable classification matrix is also added. The figure is obtained from [Dosovitskiy et al., 2020].

With the fiery success of Transformer in text dealing, Vision Transformer (ViT) [Dosovitskiy et al., 2020] is proposed in computer vision. The ViT model is inspired from Transformer [Vaswani et al., 2017], which is actually a Multilayer perceptron

(MLP) with attention blocks. There are no recurrence and convolutions but attention mechanism in this model. The core idea of ViT is Self Attention and Multi-Head Attention mechanism. The principle of them has been explained in the last section. Different from Transformer, the input of ViT is patches of images. An image is split into certain number of patches. The position information of each patch is added in each patch. A learnable matrix is also added for final classification. In every Transformer Encoder, the occurrence of Layer Normalization is to normalize the optimization space and accelerate convergence. Besides, residual Networks are added in the encoder to prevent from network degradation problems [He et al., 2016a]. MLP unit includes linear transformation with ReLU activation functions.

Regression ViT

In this work, we use ViT to directly predict the volume of cardiac structures, which is a regression task. To achieve this goal, we change the last activation function from ReLU in MLP Head into linear activation function. We use regression loss function such as MAE, MSE or Huber loss instead of loss functions for classification.

B.3.3 Experiments and analysis

Experiment protocol

ACDC dataset (182 subjects, (100×100×9) in each subject.) is used in this experiment. The dataset is split into (100, 32, 50) for training, validation, test set respectively. The training set is added with data augmentation, which is 2900 images.
For the model hyper-parameter setting, the patch size is set to 10, so the number of patches is 100. The dimension of linear projection is 128; the number of Transformer Encoder is 8; in Multi-Head Attention block, the number of heads is 4; in MLP, the transformation unit is from 256 to 128; in final MLP Head, the transformation unit is from 1024 to 512. The optimizer is AdamW [Loshchilov and Hutter, 2019] with weight decay rate 1e-4. The batchsize is 8, the learning rate is 1e-4, the training epoch is 100. The loss function is MAE, MSE and Huber loss. The model is implemented in Python with deep learning library Tensorflow 2.6.0. The model is trained from scratch for 5-fold cross-validation in a P100 GPU server.

Table B.8 – Prediction error (mean absolute error in milliliter, MAE) and error rate (PMAE, %) on volume of 3 cardiac structures using Vision Transformer with 3 different loss functions (MAE, MSE, Huber Loss=HL). ± is standard deviation.

Structure	**RV**		**MYO**		**LV**	
Model	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**	**MAE(ml)**	**PMAE(%)**
ViT_MAE	29.61±26.09	24.80±17.74	23.51±19.28	19.79±14.68	24.13±22.23	27.78±25.49
ViT_MSE	28.26±23.32	26.55±22.19	24.75±20.21	21.43±17.52	28.21±20.01	40.29±41.63
ViT_HL	28.89±24.67	26.48±20.90	23.27±20.12	20.05±17.01	28.29±19.71	42.40±44.88

Experiment results

In the computation efficiency aspect, the regression ViT model just took around 1 hour and half to train, which is faster than the regression CNNs (4 hours in Regression VGG16, 10 hours in Regression ResNet50). We tested the prediction error of ViT model under different loss functions separately. It can be found from Table B.8 that the prediction error of the three structures are average, and the all the prediction errors are slightly lower compared to regression CNNs. The author believes it has great potential to obtain better performance in this application. Because the original ViT paper [Dosovitskiy et al., 2020] also points out that the performance of ViT is higher when the dataset is of great deal. In addition of data amount, there are many hyper-parameters in the ViT model as described in the previous section which are factors to optimize the model. Thus it is necessary to keep experimenting which combination of hyper-parameters in ViT will achieve the best performance in the future works.

B.4 Conclusion

In this appendix, we added extra experiments on ACDC dataset in different aspects. For the data, we validated the input view, i.e. different number of slices, we found that the regression CNN model can not well learn the feature comprehensively from the entire slices of a cardiac, which implies that too much information may mean disruptions for a model. We also explored the influence of data augmentation at different scale, the experiments results showed that sufficient training data is good to the model's performance. For the model level, we explored different hyper parameters such as batchsize and learning rate to find suitable ones. For the dataset splitting, several dataset splitting ways with respect to ratio of training, validation and test set were tried on the limited number of medical dataset. Besides, we dis-

tributed the same number of patients according to different pathologies in training, validation and test set in order to ensure the model's generalization ability. At last, the Vision Transformer model is investigated for predicting the volume of cardiac structures. The experimental results demonstrate that the potential in regression ViT model is existed and can be explored further in the future works.

Bibliography

[Abraham and Khan, 2019] Abraham, N. and Khan, N. M. (2019). A novel focal tversky loss function with improved attention u-net for lesion segmentation. In 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pages 683–687. IEEE.

[Adams and Bischof, 1994] Adams, R. and Bischof, L. (1994). Seeded region growing. IEEE Transactions on pattern analysis and machine intelligence, 16(6):641–647.

[Adebayo et al., 2018a] Adebayo, J., Gilmer, J., Goodfellow, I., and Kim, B. (2018a). Local explanation methods for deep neural networks lack sensitivity to parameter values. In International Conference on Learning Representations (ICLR 2018 Workshop).

[Adebayo et al., 2018b] Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018b). Sanity checks for saliency maps. Advances in Neural Information Processing Systems, 31.

[Afshin et al., 2012] Afshin, M., Ayed, I. B., Islam, A., Goela, A., Peters, T. M., and Li, S. (2012). Global assessment of cardiac function using image statistics in mri. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 535–543. Springer.

[Afshin et al., 2013] Afshin, M., Ayed, I. B., Punithakumar, K., Law, M., Islam, A., Goela, A., Peters, T., and Li, S. (2013). Regional assessment of cardiac left ventricular myocardial function via mri statistical features. IEEE transactions on medical imaging, 33(2):481–494.

[Ahn et al., 2014] Ahn, B., Park, J., and Kweon, I. S. (2014). Real-time head orientation from a monocular camera using deep neural network. In Asian conference on computer vision, pages 82–96. Springer.

[Alber et al., 2019] Alber, M., Lapuschkin, S., Seegerer, P., Hägele, M., Schütt, K. T., Montavon, G., Samek, W., Müller, K.-R., Dähne, S., and Kindermans, P.-J. (2019). innvestigate neural networks! Journal of Machine Learning Research, 20(93):1–8.

[Allan, 2019] Allan, C. (2019). Melanoma Overview. `https://www.skincancer.org/skin-cancer-information/melanoma/` Accessed: October 15, 2019.

[Anders et al., 2019] Anders, C. J., Marinč, T., Neumann, D., Samek, W., Müller, K.-R., and Lapuschkin, S. (2019). Analyzing imagenet with spectral relevance analysis: Towards imagenet un-hans' ed. arXiv preprint arXiv:1912.11425.

[Arya et al., 2019] Arya, V., Bellamy, R. K. E., Chen, P.-Y., Dhurandhar, A., Hind, M., Hoffman, S. C., Houde, S., Liao, Q. V., Luss, R., Mojsilovic, A., Mourad, S., Pedemonte, P., Raghavendra, R., Richards, J. T., Sattigeri, P., Shanmugam, K., Singh, M., Varshney, K. R., Wei, D., and Zhang, Y. (2019). One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. CoRR, abs/1909.03012.

[Attia et al., 2017] Attia, M., Hossny, M., Nahavandi, S., and Yazdabadi, A. (2017). Skin melanoma segmentation using recurrent and convolutional neural networks. In IEEE ISBI, pages 292–296.

[Avendi et al., 2017] Avendi, M. R., Kheradvar, A., and Jafarkhani, H. (2017). Automatic segmentation of the right ventricle from cardiac mri using a learning-based approach. Magnetic resonance in medicine, 78(6):2439–2448.

[Bach et al., 2015] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7).

[Badrinarayanan et al., 2017] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495.

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

[Balduzzi et al., 2017] Balduzzi, D., McWilliams, B., and Butler-Yeoman, T. (2017). Neural taylor approximations: Convergence and exploration in rectifier networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 351–360. JMLR. org.

[Barnard et al., 2001] Barnard, R. W., Pearce, K., and Schovanec, L. (2001). Inequalities for the perimeter of an ellipse. Journal of mathematical analysis and applications, 260(2):295–306.

[Becker et al., 2018] Becker, S., Ackermann, M., Lapuschkin, S., Müller, K.-R., and Samek, W. (2018). Interpreting and explaining deep neural networks for classification of audio signals. arXiv preprint arXiv:1807.03418.

[Berman et al., 2018] Berman, M., Triki, A. R., and Blaschko, M. B. (2018). The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4413–4421.

[Bernard et al., 2018] Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.-A., Cetin, I., Lekadir, K., Camara, O., Ballester, M. A. G., et al. (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.

[Berthelot et al., 2019] Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., and Raffel, C. A. (2019). Mixmatch: A holistic approach to semi-supervised learning. Advances in Neural Information Processing Systems, 32.

[Brosch et al., 2015] Brosch, T., Yoo, Y., Tang, L. Y. W., Li, D. K. B., Traboulsee, A., and Tam, R. (2015). Deep convolutional encoder networks for multiple sclerosis lesion segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., editors, International Conference on Medical image computing and computer-assisted intervention (MICCAI) 2015, pages 3–11, Cham. Springer International Publishing.

[Budd et al., 2019] Budd, S., Sinclair, M., Khanal, B., Matthew, J., Lloyd, D., Gomez, A., Toussaint, N., Robinson, E. C., and Kainz, B. (2019). Confident head circumference measurement from ultrasound with real-time feedback for sonographers. In International Conference on Medical image computing and computer-assisted intervention (MICCAI), pages 683–691.

[Califf, 2018] Califf, R. M. (2018). Biomarker definitions and their applications. Experimental Biology and Medicine, 243(3):213–221.

[Caliva et al., 2019] Caliva, F., Iriondo, C., Martinez, A. M., Majumdar, S., and Pedoia, V. (2019). Distance map loss penalty term for semantic segmentation. In International Conference on Medical Imaging with Deep Learning–Extended Abstract Track.

[Canny, 1986] Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, N/A(6):679–698.

[Caruana, 1997] Caruana, R. (1997). Multitask learning. Machine learning, 28(1):41–75.

[Castro et al., 2009] Castro, J., Gómez, D., and Tejada, J. (2009). Polynomial calculation of the shapley value based on sampling. Computers & Operations Research, 36(5):1726–1730.

[Chai et al., 2021] Chai, J., Zeng, H., Li, A., and Ngai, E. W. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. Machine Learning with Applications, page 100134.

[Chang et al., 2019] Chang, C.-H., Creager, E., Goldenberg, A., and Duvenaud, D. (2019). Explaining image classifiers by counterfactual generation. In International Conference on Learning Representations.

[Chapelle et al., 2009] Chapelle, O., Scholkopf, B., and Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. IEEE Transactions on Neural Networks, 20(3):542–542.

[Chattopadhay et al., 2018] Chattopadhay, A., Sarkar, A., Howlader, P., and Balasubramanian, V. N. (2018). Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 839–847. IEEE.

[Chaudhary, 2020] Chaudhary, A. (2020). Semi-supervised learning in computer vision. `https://amitness.com/2020/07/semi-supervised-learning/` Accessed: 2021-11-22.

[Chaurasia and Culurciello, 2017] Chaurasia, A. and Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. In 2017 IEEE Visual Communications and Image Processing (VCIP), pages 1–4. IEEE.

[Chen et al., 2020a] Chen, C., Qin, C., Qiu, H., Tarroni, G., Duan, J., Bai, W., and Rueckert, D. (2020a). Deep learning for cardiac image segmentation: a review. Frontiers in Cardiovascular Medicine, 7:25.

[Chen et al., 2021a] Chen, H., Lundberg, S., and Lee, S.-I. (2021a). Explaining models by propagating shapley values of local components. In Explainable AI in Healthcare and Medicine, pages 261–270. Springer.

[Chen et al., 2016] Chen, H., Wang, X., and Heng, P. A. (2016). Automated mitosis detection with deep regression networks. In 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI), pages 1204–1207. IEEE.

[Chen et al., 2021b] Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., and Zhou, Y. (2021b). Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306.

[Chen et al., 2017] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4):834–848.

[Chen and Pavlidis, 1979] Chen, P. C. and Pavlidis, T. (1979). Segmentation by texture using a co-occurrence matrix and a split-and-merge algorithm. Computer graphics and image processing, 10(2):172–182.

[Chen et al., 2020b] Chen, R., Xu, C., Dong, Z., Liu, Y., and Du, X. (2020b). Deepcq: Deep multi-task conditional quantification network for estimation of left ventricle parameters. Computer methods and programs in biomedicine, 184:105288.

[Chollet, 2017] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1251–1258.

[Chung et al., 2014] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In NIPS 2014 Workshop on Deep Learning, December 2014.

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

[Codella et al., 2018] Codella, N. C., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., et al. (2018). Skin lesion analysis toward melanoma detection: A challenge at 2017 isbi, hosted by the international skin imaging collaboration (isic). In ISBI), pages 168–172.

[Codella et al., 2017] Codella, N. C. F., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N. K., Kittler, H., and Halpern, A. (2017). Skin lesion analysis toward melanoma detection: A challenge at 2017 isbi, hosted by the (ISIC). CoRR, abs/1710.05006.

[Dangi et al., 2018] Dangi, S., Yaniv, Z., and Linte, C. A. (2018). Left ventricle segmentation and quantification from cardiac cine mr images via multi-task learning. In International Workshop on Statistical Atlases and Computational Models of the Heart, pages 21–31. Springer.

[de La Torre et al., 2018] de La Torre, J., Puig, D., and Valls, A. (2018). Weighted kappa loss function for multi-class classification of ordinal data in deep learning. Pat Recog Let, 105:144–154.

[Decazes et al., 2019] Decazes, P., Tonnelet, D., Vera, P., and Gardin, I. (2019). Anthropometer3d: automatic multi-slice segmentation software for the measurement of anthropometric parameters from ct of pet/ct. Journal of digital imaging, 32(2):241–250.

[Degrave et al., 2016] Degrave, J., Burms, J., Korshunova, I., and Dambre, J. (2016). Using deep learning to estimate systolic and diastolic volumes from mri-images. In 25th Belgian-Dutch Conference on Machine Learning (Benelearn).

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. Ieee.

[Dhamdhere et al., 2019] Dhamdhere, K., Sundararajan, M., and Yan, Q. (2019). How important is a neuron? In International Conference on Learning Representations.

[Dice, 1945] Dice, L. R. (1945). Measures of the amount of ecologic association between species. Ecology, 26(3):297–302.

[Dobrescu et al., 2019] Dobrescu, A., Valerio Giuffrida, M., and Tsaftaris, S. A. (2019). Understanding deep neural networks for regression in leaf counting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pages 0–0.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[Du et al., 2018] Du, X., Tang, R., Yin, S., Zhang, Y., and Li, S. (2018). Direct segmentation-based full quantification for left ventricle via deep multi-task regression learning network. IEEE journal of biomedical and health informatics, 23(3):942–948.

[Elenberg et al., 2017] Elenberg, E., Dimakis, A. G., Feldman, M., and Karbasi, A. (2017). Streaming weak submodularity: Interpreting neural networks on the fly. Advances in Neural Information Processing Systems, 30.

[Esmaeili and Marvasti, 2019] Esmaeili, A. and Marvasti, F. (2019). A novel approach to quantized matrix completion using huber loss measure. IEEE Signal Processing Letters, 26(2):337–341.

[Fernández, 2021] Fernández, E. (2021). Segmentation-free estimation of aortic diameters from mri using deep learning. In Statistical Atlases and Computational Models of the Heart. M&Ms and EMIDEC Challenges: 11th International Workshop, STACOM 2020, Held in Conjunction with MICCAI 2020, Lima, Peru,

October 4, 2020, Revised Selected Papers, volume 12592, page 166. Springer Nature.

[Fiorentino et al., 2021] Fiorentino, M. C., Moccia, S., Capparuccini, M., Giamberini, S., and Frontoni, E. (2021). A regression framework to head-circumference delineation from us fetal images. Computer Methods and Programs in Biomedicine, 198:105771.

[Fong et al., 2019] Fong, R., Patrick, M., and Vedaldi, A. (2019). Understanding deep networks via extremal perturbations and smooth masks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).

[Fong and Vedaldi, 2017] Fong, R. and Vedaldi, A. (2017). Interpretable Explanations of Black Boxes by Meaningful Perturbation. 2017 IEEE International Conference on Computer Vision (ICCV), pages 3449–3457. arXiv: 1704.03296.

[Ge et al., 2019a] Ge, R., Yang, G., Chen, Y., Luo, L., Feng, C., Ma, H., Ren, J., and Li, S. (2019a). K-net: Integrate left ventricle segmentation and direct quantification of paired echo sequence. IEEE transactions on medical imaging, 39(5):1690–1702.

[Ge et al., 2019b] Ge, R., Yang, G., Chen, Y., Luo, L., Feng, C., Zhang, H., and Li, S. (2019b). Pv-lvnet: Direct left ventricle multitype indices estimation from 2d echocardiograms of paired apical views with deep neural networks. Medical image analysis, 58:101554.

[Ge et al., 2019c] Ge, R., Yang, G., Xu, C., Zhang, J., Chen, Y., Luo, L., Feng, C., Zhang, H., and Li, S. (2019c). Echoquan-net: Direct quantification of echo sequence for left ventricle multidimensional indices via global-local learning, geometric adjustment and multi-target relation learning. In International Conference on Artificial Neural Networks, pages 219–230. Springer.

[Gessert and Schlaefer, 2019] Gessert, N. and Schlaefer, A. (2019). Left ventricle quantification using direct regression with segmentation regularization and ensembles of pretrained 2d and 3d cnns. In International Workshop on Statistical Atlases and Computational Models of the Heart, pages 375–383. Springer.

[Goebel et al., 2018] Goebel, R., Chander, A., Holzinger, K., Lecue, F., Akata, Z., Stumpf, S., Kieseberg, P., and Holzinger, A. (2018). Explainable ai: the new 42?

In International cross-domain conference for machine learning and knowledge extraction, pages 295–303. Springer.

[Graves et al., 2014] Graves, A., Wayne, G., and Danihelka, I. (2014). Neural turing machines. arXiv preprint arXiv:1410.5401.

[Graziani et al., 2018] Graziani, M., Andrearczyk, V., and Müller, H. (2018). Regression concept vectors for bidirectional explanations in histopathology. In Understanding and Interpreting Machine Learning in Medical Image Computing Applications, pages 124–132. Springer.

[Gu et al., 2018] Gu, B., Shan, Y., Sheng, V. S., Zheng, Y., and Li, S. (2018). Sparse regression with output correlation for cardiac ejection fraction estimation. Information Sciences, 423:303–312.

[Gu and Tresp, 2019] Gu, J. and Tresp, V. (2019). Contextual prediction difference analysis for explaining individual image classifications. arXiv preprint arXiv:1910.09086.

[Guidotti et al., 2018] Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., and Giannotti, F. (2018). Local rule-based explanations of black box decision systems. arXiv preprint arXiv:1805.10820.

[Guo et al., 2019] Guo, Y., Shi, H., Kumar, A., Grauman, K., Rosing, T., and Feris, R. (2019). Spottune: transfer learning through adaptive fine-tuning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4805–4814.

[Gutman et al., 2016] Gutman, D., Codella, N. C. F., Celebi, M. E., Helba, B., Marchetti, M. A., Mishra, N. K., and Halpern, A. (2016). Skin lesion analysis toward melanoma detection: A challenge at the international symposium on biomedical imaging (ISBI) 2016, hosted by the international skin imaging collaboration (ISIC). CoRR, abs/1605.01397.

[Hartigan and Wong, 1979] Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. Journal of the royal statistical society. series c (applied statistics), 28(1):100–108.

[Hashemi et al., 2018] Hashemi, S. R., Salehi, S. S. M., Erdogmus, D., Prabhu, S. P., Warfield, S. K., and Gholipour, A. (2018). Asymmetric loss functions and deep

densely-connected networks for highly-imbalanced medical image segmentation: Application to multiple sclerosis lesion detection. IEEE Access, 7:1721–1735.

[Hassanzadeh et al., 2020] Hassanzadeh, T., Essam, D., and Sarker, R. (2020). 2d to 3d evolutionary deep convolutional neural networks for medical image segmentation. IEEE Transactions on Medical Imaging, 40(2):712–721.

[He et al., 2016a] He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

[He et al., 2016b] He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Identity mappings in deep residual networks. In European conference on computer vision, pages 630–645. Springer.

[He et al., 2019] He, T., Guo, J., Wang, J., Xu, X., and Yi, Z. (2019). Multi-task learning for the segmentation of thoracic organs at risk in ct images. In SegTHOR@ ISBI, pages 10–13.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8):1735–1780.

[Howard et al., 2017] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

[Hu et al., 2018] Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7132–7141.

[Huang et al., 2017] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4700–4708.

[Huang et al., 2021a] Huang, S.-G., Chung, M. K., Qiu, A., Initiative, A. D. N., et al. (2021a). Fast mesh data augmentation via chebyshev polynomial of spectral filtering. Neural Networks.

[Huang et al., 2021b] Huang, X., Tian, Y., Zhao, S., Liu, T., Wang, W., and Wang, Q. (2021b). Direct full quantification of the left ventricle via multitask regression and classification. Applied Intelligence, pages 1–14.

[Hubert, 1977] Hubert, L. (1977). Kappa revisited. Psychological Bulletin, 84(2):289.

[Hussain et al., 2016] Hussain, M. A., Hamarneh, G., O'Connell, T. W., Mohammed, M. F., and Abugharbieh, R. (2016). Segmentation-free estimation of kidney volumes in ct with dual regression forests. In International Workshop on Machine Learning in Medical Imaging, pages 156–163. Springer.

[Indyk and Motwani, 1998] Indyk, P. and Motwani, R. (1998). Approximate nearest neighbors: towards removing the curse of dimensionality. In Proceedings of the thirtieth annual ACM symposium on Theory of computing, pages 604–613.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR.

[Isensee et al., 2017] Isensee, F., Jaeger, P. F., Full, P. M., Wolf, I., Engelhardt, S., and Maier-Hein, K. H. (2017). Automatic cardiac disease assessment on cine-mri via time-series segmentation and domain specific features. In International workshop on statistical atlases and computational models of the heart, pages 120–129. Springer.

[Ivanov et al., 2019] Ivanov, I., Lomaev, Y., and Barkovskaya, A. (2019). Automatic calculation of left ventricular volume in magnetic resonance imaging using an image-based clustering approach. IOP Conference Series: Materials Science and Engineering, 537(4):042046.

[Jafari et al., 2016] Jafari, M. H., Karimi, N., Nasr-Esfahani, E., Samavi, S., Soroushmehr, S. M. R., Ward, K., and Najarian, K. (2016). Skin lesion segmentation in clinical images using deep learning. In IEEE ICPR, pages 337–342.

[Jain et al., 2015] Jain, S., Pise, N., et al. (2015). Computer aided melanoma skin cancer detection using image processing. Procedia Computer Science, 48:735–740.

[Jang et al., 2017] Jang, Y., Hong, Y., Ha, S., Kim, S., and Chang, H.-J. (2017). Automatic segmentation of lv and rv in cardiac mri. In International Workshop

on Statistical Atlases and Computational Models of the Heart, pages 161–169. Springer.

[Jardim and Figueiredo, 2005] Jardim, S. M. and Figueiredo, M. A. (2005). Segmentation of fetal ultrasound images. Ultrasound in Medicine and Biology, 31(2):243 – 250.

[Jha et al., 2020] Jha, D., Riegler, M. A., Johansen, D., Halvorsen, P., and Johansen, H. D. (2020). Doubleu-net: A deep convolutional neural network for medical image segmentation. In 2020 IEEE 33rd International symposium on computer-based medical systems (CBMS), pages 558–564. IEEE.

[Jia et al., 2021] Jia, J., Zhai, Z., Bakker, M. E., Hernández-Girón, I., Staring, M., and Stoel, B. C. (2021). Multi-task semi-supervised learning for pulmonary lobe segmentation. In 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pages 1329–1332. IEEE.

[Jia et al., 2018] Jia, S., Despinasse, A., Wang, Z., Delingette, H., Pennec, X., Jaïs, P., Cochet, H., and Sermesant, M. (2018). Automatically segmenting the left atrium from cardiac images using successive 3d u-nets and a contour loss. In Workshop on Statistical Atlases and Computational Models of the Heart, pages 221–229. Springer.

[Karimi and Salcudean, 2019] Karimi, D. and Salcudean, S. E. (2019). Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. IEEE Transactions on medical imaging, 39(2):499–513.

[Kass et al., 1988] Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. International journal of computer vision, 1(4):321–331.

[Kervadec et al., 2019] Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., and Ayed, I. B. (2019). Boundary loss for highly unbalanced segmentation. In International conference on medical imaging with deep learning, pages 285–296. PMLR.

[Kim et al., 2018] Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., and sayres, R. (2018). Interpretability beyond feature attribution: Quantitative

testing with concept activation vectors (TCAV). In Dy, J. and Krause, A., editors, Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pages 2668–2677. PMLR.

[Kim et al., 2019] Kim, H. P., Lee, S. M., Kwon, J.-Y., Park, Y., Kim, K. C., and Seo, J. K. (2019). Automatic evaluation of fetal head biometry from ultrasound images using machine learning. Physiological measurement, 40(6):065009.

[Kindermans et al., 2019] Kindermans, P.-J., Hooker, S., Adebayo, J., Alber, M., Schütt, K. T., Dähne, S., Erhan, D., and Kim, B. (2019). The (un) reliability of saliency methods. In Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, pages 267–280. Springer.

[Kindermans et al., 2018] Kindermans, P.-J., Schütt, K. T., Alber, M., Müller, K.-R., Erhan, D., Kim, B., and Dähne, S. (2018). Learning how to explain neural networks: Patternnet and patternattribution. In International Conference on Learning Representations.

[Kingma and Ba, 2014] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. International Conference on Learning Representations.

[Kong et al., 2016] Kong, B., Zhan, Y., Shin, M., Denny, T., and Zhang, S. (2016). Recognizing end-diastole and end-systole frames via deep temporal regression network. In International conference on medical image computing and computer-assisted intervention, pages 264–272. Springer.

[Krizhevsky et al., 2009] Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

[Kumar et al., 2014] Kumar, V., Abbas, A. K., Fausto, N., and Aster, J. C. (2014). Robbins and Cotran pathologic basis of disease, professional edition e-book. Elsevier health sciences.

[Kwee and Kwee, 2020] Kwee, T. C. and Kwee, R. M. (2020). Chest ct in covid-19: what the radiologist needs to know. RadioGraphics, 40(7):1848–1865.

[Laine and Aila, 2017] Laine, S. and Aila, T. (2017). Temporal ensembling for semi-supervised learning. In International Conference on Learning Representations (ICLR 2017 Workshop).

[Lathuilière et al., 2019] Lathuilière, S., Mesejo, P., Alameda-Pineda, X., and Horaud, R. (2019). A comprehensive analysis of deep regression. IEEE transactions on pattern analysis and machine intelligence.

[Leclerc et al., 2019] Leclerc, S., Smistad, E., Pedrosa, J., Østvik, A., Cervenansky, F., Espinosa, F., Espeland, T., Berg, E. A. R., Jodoin, P.-M., Grenier, T., et al. (2019). Deep learning for segmentation using an open large-scale dataset in 2d echocardiography. IEEE transactions on medical imaging, 38(9):2198–2210.

[Lecue, 2020] Lecue, F. (2020). On the role of knowledge graphs in explainable ai. Semantic Web, 11(1):41–51.

[Lee et al., 2013] Lee, D.-H. et al. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. Workshop on challenges in representation learning, ICML, 3(2):896.

[Lee, 2010] Lee, J. S. (2010). Technical advances in current pet and hybrid imaging systems. The Open Nuclear Medicine Journal, 2(1).

[Leino et al., 2018] Leino, K., Sen, S., Datta, A., Fredrikson, M., and Li, L. (2018). Influence-directed explanations for deep convolutional networks. In 2018 IEEE International Test Conference (ITC), pages 1–8. IEEE.

[Li et al., 2016] Li, J., Monroe, W., and Jurafsky, D. (2016). Understanding neural networks through representation erasure. arXiv preprint arXiv:1612.08220.

[Li et al., 2017] Li, J., Wang, Y., Lei, B., Cheng, J.-Z., Qin, J., Wang, T., Li, S., and Ni, D. (2017). Automatic fetal head circumference measurement in ultrasound using random forest and fast ellipse fitting. IEEE journal of biomedical and health informatics, 22(1):215–223.

[Li et al., 2020] Li, T., Wei, B., Cong, J., Hong, Y., and Li, S. (2020). Direct estimation of left ventricular ejection fraction via a cardiac cycle feature learning architecture. Computers in biology and medicine, 118:103659.

[Lian et al., 2021] Lian, C., Liu, M., Wang, L., and Shen, D. (2021). Multi-task weakly-supervised attention network for dementia status estimation with structural mri. IEEE Transactions on Neural Networks and Learning Systems.

[Liao et al., 2017] Liao, F., Chen, X., Hu, X., and Song, S. (2017). Estimation of the volume of the left ventricle from mri images using deep neural networks. IEEE transactions on cybernetics, 49(2):495–504.

[Lin et al., 2017a] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2117–2125.

[Lin et al., 2017b] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pages 2980–2988.

[Liu et al., 2019] Liu, H., Yin, Q., and Wang, W. Y. (2019). Towards explainable nlp: A generative explanation framework for text classification. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5570–5581.

[Liu et al., 2018] Liu, J., Li, X., Ren, H., and Li, Q. (2018). Multi-estimator full left ventricle quantification through ensemble learning. In International Workshop on Statistical Atlases and Computational Models of the Heart, pages 459–465. Springer.

[Liu et al., 2021a] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021a). Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 10012–10022.

[Liu et al., 2021b] Liu, Z., Lu, Y., Zhang, X., Wang, S., Li, S., and Chen, B. (2021b). Multi-indices quantification for left ventricle via densenet and gru-based encoder-decoder with attention. Complexity, 2021.

[Liu et al., 2022] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). A convnet for the 2020s. arXiv preprint arXiv:2201.03545.

[Liu et al., 2020] Liu, Z., Zhang, Y., Li, W., Li, S., Zou, Z., and Chen, B. (2020). Multislice left ventricular ejection fraction prediction from cardiac mris without segmentation using shared sptdennet. Computerized Medical Imaging and Graphics, 86:101795.

[Long et al., 2015] 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.

[Loshchilov and Hutter, 2019] Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In International Conference on Learning Representations (ICLR 2019).

[Lu et al., 2005] Lu, W., Tan, J., and Floyd, R. (2005). Automated fetal head detection and measurement in ultrasound images by iterative randomized hough transform. Ultrasound in Medicine & Biology, 31(7):929 – 936.

[Lundberg and Lee, 2017] Lundberg, S. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. arXiv preprint arXiv:1705.07874.

[Lundberg et al., 2020] Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable ai for trees. Nature machine intelligence, 2(1):56–67.

[Luo et al., 2017] Luo, G., Dong, S., Wang, K., Zuo, W., Cao, S., and Zhang, H. (2017). Multi-views fusion cnn for left ventricular volumes estimation on cardiac mr images. IEEE Transactions on Biomedical Engineering, 65(9):1924–1934.

[Luo et al., 2020a] Luo, G., Dong, S., Wang, W., Wang, K., Cao, S., Tam, C., Zhang, H., Howey, J., Ohorodnyk, P., and Li, S. (2020a). Commensal correlation network between segmentation and direct area estimation for bi-ventricle quantification. Medical image analysis, 59:101591.

[Luo et al., 2016] Luo, G., Sun, G., Wang, K., Dong, S., and Zhang, H. (2016). A novel left ventricular volumes prediction method based on deep learning network in cardiac mri. In 2016 Computing in Cardiology Conference (CinC), pages 89–92. IEEE.

[Luo et al., 2019] Luo, G., Wang, K., Wulan, N., Cao, S., Li, Q., Yuan, Y., and Zhang, H. (2019). A novel spatio-temporal self-supervised framework to improve the generalization ability for left ventricle volume quantification based on cmr data. In 2019 Computing in Cardiology (CinC), pages Page–1. IEEE.

[Luo et al., 2020b] Luo, G., Wang, W., Tam, C., Wang, K., Cao, S., Zhang, H., Chen, B., and Li, S. (2020b). Dynamically constructed network with error correction for accurate ventricle volume estimation. Medical Image Analysis, 64:101723.

[Luong et al., 2015] Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In Conference on Empirical Methods in Natural Language Processing (EMNLP).

[Ma et al., 2021] Ma, J., Chen, J., Ng, M., Huang, R., Li, Y., Li, C., Yang, X., and Martel, A. L. (2021). Loss odyssey in medical image segmentation. Medical Image Analysis, 71:102035.

[Malladi et al., 1995] Malladi, R., Sethian, J. A., and Vemuri, B. C. (1995). Shape modeling with front propagation: A level set approach. IEEE transactions on pattern analysis and machine intelligence, 17(2):158–175.

[Melamed et al., 2011] Melamed, N., Yogev, Y., Danon, D., Mashiach, R., Meizner, I., and Ben-Haroush, A. (2011). Sonographic estimation of fetal head circumference: how accurate are we? Ultrasound in obstetrics & gynecology, 37(1):65–71.

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

[Miyato et al., 2018] Miyato, T., Maeda, S.-i., Koyama, M., and Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE transactions on pattern analysis and machine intelligence, 41(8):1979–1993.

[Moccia et al., 2021] Moccia, S., Fiorentino, M. C., and Frontoni, E. (2021). Mask-r^2 cnn: a distance-field regression version of mask-rcnn for fetal-head delineation in ultrasound images. International Journal of Computer Assisted Radiology and Surgery, pages 1–8.

[Molnar, 2019] Molnar, C. (2019). Interpretable Machine Learning. A Guide for Making Black Box Models Explainable. https://christophm.github.io/interpretable-ml-book/. Accessed: 2020-5-10.

[Montavon et al., 2017] Montavon, G., Lapuschkin, S., Binder, A., Samek, W., and Müller, K.-R. (2017). Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition, 65:211–222.

[Morch et al., 1995] Morch, N. J., Kjems, U., Hansen, L. K., Svarer, C., Law, I., Lautrup, B., Strother, S., and Rehm, K. (1995). Visualization of neural networks using saliency maps. In Proceedings of IEEE International Conference on Neural Networks, volume 4, pages 2085–2090.

[Mordvintsev et al., 2015] Mordvintsev, A., Olah, C., and Tyka, M. (2015). Inceptionism: Going deeper into neural networks. Google Research Blog.

[Ngo et al., 2017] Ngo, T. A., Lu, Z., and Carneiro, G. (2017). Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance. Medical image analysis, 35:159–171.

[Oktay et al., 2018] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M. J., Heinrich, M. P., Misawa, K., Mori, K., McDonagh, S. G., Hammerla, N. Y., Kainz, B., Glocker, B., and Rueckert, D. (2018). Attention u-net: Learning where to look for the pancreas. In Medical Imaging with Deep Learning (MIDL).

[Otsu, 1979] Otsu, N. (1979). A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics, 9(1):62–66.

[O'Mahony et al., 2019] O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., and Walsh, J. (2019). Deep learning vs. traditional computer vision. In Science and Information Conference, pages 128–144. Springer.

[Pang et al., 2018] Pang, S., Leung, S., Nachum, I. B., Feng, Q., and Li, S. (2018). Direct automated quantitative measurement of spine via cascade amplifier regression network. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 940–948. Springer.

[Pang et al., 2019] Pang, S., Su, Z., Leung, S., Nachum, I. B., Chen, B., Feng, Q., and Li, S. (2019). Direct automated quantitative measurement of spine by cascade amplifier regression network with manifold regularization. Medical image analysis, 55:103–115.

[Pedemonte et al., 2018] Pedemonte, S., Bizzo, B., Pomerantz, S., Tenenholtz, N., Wright, B., Walters, M., Doyle, S., McCarthy, A., De Almeida, R. R., Andriole, K., et al. (2018). Detection and delineation of acute cerebral infarct on dwi using weakly supervised machine learning. In International Conference on Medical image computing and computer-assisted intervention (MICCAI), pages 81–88.

[Pennisi et al., 2016] Pennisi, A., Bloisi, D. D., Nardi, D., Giampetruzzi, A. R., Mondino, C., and Facchiano, A. (2016). Skin lesion image segmentation using delaunay triangulation for melanoma detection. Computerized Medical Imaging and Graphics, 52:89 – 103.

[Pereira et al., 2020] Pereira, R. F., Rebelo, M. S., Moreno, R. A., Marco, A. G., Lima, D. M., Arruda, M. A., Krieger, J. E., and Gutierrez, M. A. (2020). Fully automated quantification of cardiac indices from cine mri using a combination of convolution neural networks. In 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), pages 1221–1224. IEEE.

[Petitjean and Dacher, 2011] Petitjean, C. and Dacher, J.-N. (2011). A review of segmentation methods in short axis cardiac mr images. Medical image analysis, 15(2):169–184.

[Petsiuk et al., 2018] Petsiuk, V., Das, A., and Saenko, K. (2018). Rise: Randomized input sampling for explanation of black-box models. In British Machine Vision Conference (BMVC).

[Raghunath et al., 2020] Raghunath, S., Cerna, A. E. U., Jing, L., Stough, J., Hartzel, D. N., Leader, J. B., Kirchner, H. L., Stumpe, M. C., Hafez, A., Nemani, A., et al. (2020). Prediction of mortality from 12-lead electrocardiogram voltage data using a deep neural network. Nature medicine, 26(6):886–891.

[Rahman and Wang, 2016] Rahman, M. A. and Wang, Y. (2016). Optimizing intersection-over-union in deep neural networks for image segmentation. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Porikli, F., Skaff, S., Entezari, A., Min, J., Iwai, D., Sadagic, A., Scheidegger, C., and Isenberg, T., editors, Advances in Visual Computing, pages 234–244, Cham. Springer International Publishing.

[Ramesh et al., 2021] Ramesh, K., Kumar, G. K., Swapna, K., Datta, D., and Rajest, S. S. (2021). A review of medical image segmentation algorithms. EAI Endorsed Transactions on Pervasive Health and Technology.

[Ribeiro et al., 2016] 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.

[Ribeiro et al., 2018] Ribeiro, M. T., Singh, S., and Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1).

[Ribera et al., 2019] Ribera, J., Guera, D., Chen, Y., and Delp, E. J. (2019). Locating objects without bounding boxes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6479–6489.

[Riegler et al., 2013] Riegler, G., Ferstl, D., Rüther, M., and Bischof, H. (2013). Hough networks for head pose estimation and facial feature localization. Journal of Computer Vision, 101(3):437–458.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (MICCAI), pages 234–241. Springer.

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004). " grabcut" interactive foreground extraction using iterated graph cuts. ACM transactions on graphics (TOG), 23(3):309–314.

[Rudin, 2019] Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence, 1(5):206–215.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088):533–536.

[Salehi et al., 2017] Salehi, S. S. M., Erdogmus, D., and Gholipour, A. (2017). Tversky loss function for image segmentation using 3d fully convolutional deep networks. In International workshop on machine learning in medical imaging, pages 379–387. Springer.

[Samek et al., 2016] Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K.-R. (2016). Evaluating the visualization of what a deep neural network has learned. IEEE transactions on neural networks and learning systems, 28(11):2660–2673.

[Samek et al., 2021] Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., and Müller, K.-R. (2021). Explaining deep neural networks and beyond: A review of methods and applications. Proceedings of the IEEE, 109(3):247–278.

[Sarris et al., 2012] Sarris, I., Ioannou, C., Chamberlain, P., Ohuma, E., Roseman, F., Hoch, L., Altman, D., Papageorghiou, A., and INTERGROWTH-21st (2012). Intra- and interobserver variability in fetal ultrasound measurements. Ultrasound in obstetrics & gynecology, 39(3):266–273.

[Savioli et al., 2018] Savioli, N., Vieira, M. S., Lamata, P., and Montana, G. (2018). A generative adversarial model for right ventricle segmentation. arXiv preprint arXiv:1810.03969.

[Schaefer et al., 2011] Schaefer, G., Rajab, M. I., Celebi, M. E., and Iyatomi, H. (2011). Colour and contrast enhancement for improved skin lesion segmentation. Computerized Medical Imaging and Graphics, 35(2):99–104.

[Schnake et al., 2020] Schnake, T., Eberle, O., Lederer, J., Nakajima, S., Schütt, K., Müller, K., and Montavon, G. (2020). Higher-order explanations of graph neural networks via relevant walks. arXiv: 2006.03589.

[Selvaraju et al., 2017] 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 (ICCV).

[Serge and Lantuéj, 1979] Serge, B. and Lantuéj, C. (1979). Use of watersheds in contour detection. In Workshop on Image Processing, Real-time Edge and Motion Detection, Rennes, France.

[Shrikumar et al., 2017] Shrikumar, A., Greenside, P., and Kundaje, A. (2017). Learning important features through propagating activation differences. In Precup, D. and Teh, Y. W., editors, Proceedings of the 34th International Conference on

Machine Learning, volume 70 of Proceedings of Machine Learning Research, pages 3145–3153. PMLR.

[Shrikumar et al., 2016] Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. arXiv preprint arXiv:1605.01713.

[Shrikumar et al., 2018] Shrikumar, A., Su, J., and Kundaje, A. (2018). Computationally efficient measures of internal neuron importance. CoRR, abs/1807.09946.

[Si and Roberts, 2019] Si, Y. and Roberts, K. (2019). Deep patient representation of clinical notes via multi-task learning for mortality prediction. AMIA Summits on Translational Science Proceedings, 2019:779.

[Simonyan et al., 2013] Simonyan, K., Vedaldi, A., and Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034.

[Simonyan and Zisserman, 2015] Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations (ICLR).

[Smilkov et al., 2017] Smilkov, D., Thorat, N., Kim, B., Viégas, F. B., and Wattenberg, M. (2017). Smoothgrad: removing noise by adding noise. In Workshop on Visualization for Deep Learning, ICML.

[Sobhaninia et al., 2019] Sobhaninia, Z., Rafiei, S., Emami, A., Karimi, N., Najarian, K., Samavi, S., and Soroushmehr, S. R. (2019). Fetal ultrasound image segmentation for measuring biometric parameters using multi-task deep learning. In Conference of the IEEE EMBC, pages 6545–6548.

[Sohn et al., 2020] Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. (2020). Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in Neural Information Processing Systems, 33.

[Springenberg et al., 2015] Springenberg, J., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2015). Striving for simplicity: The all convolutional net. In International Conference on Learning Representations (ICLR).

[Sudre et al., 2017] Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., and Cardoso, M. J. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In Deep learning in medical image analysis and multimodal learning for clinical decision support, pages 240–248. Springer.

[Sun et al., 2017] Sun, H., Zhen, X., Bailey, C., Rasoulinejad, P., Yin, Y., and Li, S. (2017). Direct estimation of spinal cobb angles by structured multi-output regression. In International conference on information processing in medical imaging, pages 529–540. Springer.

[Sundararajan et al., 2017] Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 3319–3328. JMLR. org.

[Szegedy et al., 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2818–2826.

[Taghanaki et al., 2021] Taghanaki, S. A., Abhishek, K., Cohen, J. P., Cohen-Adad, J., and Hamarneh, G. (2021). Deep semantic segmentation of natural and medical images: a review. Artificial Intelligence Review, 54(1):137–178.

[Taghanaki et al., 2019] Taghanaki, S. A., Zheng, Y., Zhou, S. K., Georgescu, B., Sharma, P., Xu, D., Comaniciu, D., and Hamarneh, G. (2019). Combo loss: Handling input and output imbalance in multi-organ segmentation. Computerized Medical Imaging and Graphics, 75:24–33.

[Tan et al., 2020] Tan, C., Chen, S., Ji, G., and Geng, X. (2020). Multilabel distribution learning based on multioutput regression and manifold learning. IEEE Transactions on Cybernetics.

[Tan and Le, 2019] Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, pages 6105–6114. PMLR.

[Tao et al., 2019] Tao, Q., Yan, W., Wang, Y., Paiman, E. H., Shamonin, D. P., Garg, P., Plein, S., Huang, L., Xia, L., Sramko, M., et al. (2019). Deep learning–based

method for fully automatic quantification of left ventricle function from cine mr images: a multivendor, multicenter study. Radiology, 290(1):81–88.

[Tarvainen and Valpola, 2017] Tarvainen, A. and Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Proceedings of the 31st International Conference on Neural Information Processing Systems, pages 1195–1204.

[Tjoa and Guan, 2021] Tjoa, E. and Guan, C. (2021). A survey on explainable artificial intelligence (xai): Toward medical xai. IEEE Transactions on Neural Networks and Learning Systems, 32(11):4793–4813.

[Trullo et al., 2017] Trullo, R., Petitjean, C., Ruan, S., Dubray, B., Nie, D., and Shen, D. (2017). Segmentation of organs at risk in thoracic ct images using a sharpmask architecture and conditional random fields. In IEEE ISBI, pages 1003–1006.

[Tschandl et al., 2018] Tschandl, P., Rosendahl, C., and Kittler, H. (2018). The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific data, 5:180161.

[van den Heuvel et al., 2018a] van den Heuvel, T. L. A., de Bruijn, D., de Korte, C. L., and Ginneken, B. v. (2018a). Automated measurement of fetal head circumference using 2d ultrasound images. PLOS ONE, 13(8):1–20.

[van den Heuvel et al., 2018b] van den Heuvel, T. L. A., de Bruijn, D., de Korte, C. L., and Ginneken, B. v. (2018b). Automated measurement of fetal head circumference using 2d ultrasound images [data set]. Zenodo. Accessed: 2019-12-11.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems, pages 5998–6008.

[Vesal et al., 2020] Vesal, S., Gu, M., Maier, A., and Ravikumar, N. (2020). Spatio-temporal multi-task learning for cardiac mri left ventricle quantification. IEEE Journal of Biomedical and Health Informatics.

[Vigneault et al., 2018] Vigneault, D. M., Xie, W., Ho, C. Y., Bluemke, D. A., and Noble, J. A. (2018). ω-net (omega-net): fully automatic, multi-view cardiac mr detection, orientation, and segmentation with deep neural networks. Medical image analysis, 48:95–106.

[Vu et al., 2020] Vu, M. H., Grimbergen, G., Nyholm, T., and Löfstedt, T. (2020). Evaluation of multislice inputs to convolutional neural networks for medical image segmentation. Medical Physics, 47(12):6216–6231.

[Vuong et al., 2019] Vuong, Q.-H., Ho, M.-T., Vuong, T.-T., La, V.-P., Ho, M.-T., Nghiem, K.-C. P., Tran, B. X., Giang, H.-H., Giang, T.-V., Latkin, C., et al. (2019). Artificial intelligence vs. natural stupidity: Evaluating ai readiness for the vietnamese medical information system. Journal of clinical medicine, 8(2):168.

[Wacker et al., 2020] Wacker, J., Ladeira, M., and Nascimento, J. (2020). Transfer learning for brain tumor segmentation. In BrainLes@MICCAI.

[Wang et al., 2015] Wang, H., Shi, W., Bai, W., de Marvao, A. M. S. M., Dawes, T. J., O'Regan, D. P., Edwards, P., Cook, S., and Rueckert, D. (2015). Prediction of clinical information from cardiac mri using manifold learning. In International Conference on Functional Imaging and Modeling of the Heart, pages 91–98. Springer.

[Wang et al., 2019] Wang, W., Wang, Y., Wu, Y., Lin, T., Li, S., and Chen, B. (2019). Quantification of full left ventricular metrics via deep regression learning with contour-guidance. IEEE Access, 7:47918–47928.

[Wang et al., 2014] Wang, Z., Salah, M. B., Gu, B., Islam, A., Goela, A., and Li, S. (2014). Direct estimation of cardiac biventricular volumes with an adapted bayesian formulation. IEEE Transactions on Biomedical Engineering, 61(4):1251–1260.

[Wang and Yang, 2018] Wang, Z. and Yang, J. (2018). Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation. In Workshops at the thirty-second AAAI conference on artificial intelligence.

[Weng, 2018] Weng, L. (2018). Attention? attention! `http://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html`. Accessed: 2021-11-15.

[WHO, 2021] WHO (2021). Cardiovascular diseases. `https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)`. Accessed: 2021-11-22.

[Wikipedia, 2021] Wikipedia (2021). Medical image computing. `https://en.wikipedia.org/wiki/Medical_image_computing`. Accessed: 2021-10-06.

[Wong et al., 2011] Wong, A., Scharcanski, J., and Fieguth, P. (2011). Automatic skin lesion segmentation via iterative stochastic region merging. IEEE Transactions on Information Technology in Biomedicine, 15(6):929–936.

[Wong et al., 2018] Wong, K. C., Moradi, M., Tang, H., and Syeda-Mahmood, T. (2018). 3d segmentation with exponential logarithmic loss for highly unbalanced object sizes. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 612–619. Springer.

[Xie et al., 2020a] Xie, F., Yang, J., Liu, J., Jiang, Z., Zheng, Y., and Wang, Y. (2020a). Skin lesion segmentation using high-resolution convolutional neural network. Computer Methods and Programs in Biomedicine, 186:105241.

[Xie et al., 2020b] Xie, Q., Dai, Z., Hovy, E., Luong, T., and Le, Q. (2020b). Unsupervised data augmentation for consistency training. Advances in Neural Information Processing Systems, 33.

[Xie et al., 2020c] Xie, Q., Luong, M.-T., Hovy, E., and Le, Q. V. (2020c). Self-training with noisy student improves imagenet classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10687–10698.

[Xu et al., 2018] Xu, C., Xu, L., Brahm, G., Zhang, H., and Li, S. (2018). Mutgan: Simultaneous segmentation and quantification of myocardial infarction without contrast agents via joint adversarial learning. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 525–534. Springer.

[Xue et al., 2018] Xue, W., Brahm, G., Pandey, S., Leung, S., and Li, S. (2018). Full left ventricle quantification via deep multitask relationships learning. Medical image analysis, 43:54–65.

[Xue et al., 2017a] Xue, W., Islam, A., Bhaduri, M., and Li, S. (2017a). Direct multi-type cardiac indices estimation via joint representation and regression learning. IEEE transactions on medical imaging, 36(10):2057–2067.

[Xue et al., 2017b] Xue, W., Lum, A., Mercado, A., Landis, M., Warrington, J., and Li, S. (2017b). Full quantification of left ventricle via deep multitask learning network respecting intra-and inter-task relatedness. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 276–284. Springer.

[Xue et al., 2017c] Xue, W., Nachum, I. B., Pandey, S., Warrington, J., Leung, S., and Li, S. (2017c). Direct estimation of regional wall thicknesses via residual recurrent neural network. In International Conference on Information Processing in Medical Imaging, pages 505–516. Springer.

[Yakubovskiy, 2019] Yakubovskiy, P. (2019). Segmentation models. `https://github.com/qubvel/segmentation_models`. Accessed: 2021-4-12.

[Yang et al., 2019] Yang, H.-L., Kim, J. J., Kim, J. H., Kang, Y. K., Park, D. H., Park, H. S., Kim, H. K., and Kim, M.-S. (2019). Weakly supervised lesion localization for age-related macular degeneration detection using optical coherence tomography images. Plos one, 14(4):e0215076.

[Yang et al., 2021] Yang, J., Huang, X., He, Y., Xu, J., Yang, C., Xu, G., and Ni, B. (2021). Reinventing 2d convolutions for 3d images. IEEE Journal of Biomedical and Health Informatics, 25(8):3009–3018.

[Yang et al., 2017] Yang, X., Bian, C., Yu, L., Ni, D., and Heng, P.-A. (2017). Class-balanced deep neural network for automatic ventricular structure segmentation. In International workshop on statistical atlases and computational models of the heart, pages 152–160. Springer.

[Yeche et al., 2019] Yeche, H., Harrison, J., and Berthier, T. (2019). Ubs: A dimension-agnostic metric for concept vector interpretability applied to radiomics. In Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support, pages 12–20. Springer.

[Ying et al., 2019] Ying, R., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. (2019). Gnnexplainer: Generating explanations for graph neural networks. Advances in neural information processing systems, 32:9240.

[Yu et al., 2021] Yu, C., Gao, Z., Zhang, W., Yang, G., Zhao, S., Zhang, H., Zhang, Y., and Li, S. (2021). Multitask learning for estimating multitype cardiac indices in

mri and ct based on adversarial reverse mapping. IEEE transactions on neural networks and learning systems, 32(2):493–506.

[Yuan et al., 2009] Yuan, X., Situ, N., and Zouridakis, G. (2009). A narrow band graph partitioning method for skin lesion segmentation. Pattern Recognition, 42(6):1017–1028.

[Yuan et al., 2017] Yuan, Y., Chao, M., and Lo, Y.-C. (2017). Automatic skin lesion segmentation using deep fully convolutional networks with jaccard distance. IEEE transactions on medical imaging, 36(9):1876–1886.

[Zeiler and Fergus, 2014] Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In European conference on computer vision, pages 818–833. Springer.

[Zhang et al., 2012] Zhang, D., Shen, D., Initiative, A. D. N., et al. (2012). Multimodal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease. NeuroImage, 59(2):895–907.

[Zhang et al., 2020a] Zhang, D., Yang, G., Zhao, S., Zhang, Y., Ghista, D., Zhang, H., and Li, S. (2020a). Direct quantification of coronary artery stenosis through hierarchical attentive multi-view learning. IEEE Transactions on Medical Imaging, 39(12):4322–4334.

[Zhang et al., 2019] Zhang, D., Yang, G., Zhao, S., Zhang, Y., Zhang, H., and Li, S. (2019). Direct quantification for coronary artery stenosis using multiview learning. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 449–457. Springer.

[Zhang et al., 2018] Zhang, J., Bargal, S. A., Lin, Z., Brandt, J., Shen, X., and Sclaroff, S. (2018). Top-down neural attention by excitation backprop. International Journal of Computer Vision, 126(10):1084–1102.

[Zhang et al., 2020b] Zhang, J., Petitjean, C., Lopez, P., and Ainouz, S. (2020b). Direct estimation of fetal head circumference from ultrasound images based on regression cnn. In Medical Imaging with Deep Learning, pages 914–922. PMLR.

[Zhang et al., 2020c] Zhang, J., Petitjean, C., Yger, F., and Ainouz, S. (2020c). Explainability for regression cnn in fetal head circumference estimation from ultra-

sound images. In Interpretable and Annotation-Efficient Learning for Medical Image Computing, pages 73–82. Springer.

[Zhao et al., 2021] Zhao, C., Li, D., Feng, C., and Li, S. (2021). Of-umrn: Uncertainty-guided multitask regression network aided by optical flow for fully automated comprehensive analysis of carotid artery. Medical Image Analysis, page 101982.

[Zhao et al., 2017] Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2881–2890.

[Zhen et al., 2015a] Zhen, X., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2015a). Direct and simultaneous four-chamber volume estimation by multi-output regression. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 669–676. Springer.

[Zhen et al., 2014] Zhen, X., Wang, Z., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2014). Direct estimation of cardiac bi-ventricular volumes with regression forests. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 586–593. Springer.

[Zhen et al., 2015b] Zhen, X., Wang, Z., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2015b). Direct volume estimation without segmentation. In Medical Imaging 2015: Image Processing, volume 9413, page 94132G. International Society for Optics and Photonics.

[Zhen et al., 2016a] Zhen, X., Wang, Z., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2016a). Multi-scale deep networks and regression forests for direct bi-ventricular volume estimation. Medical image analysis, 30:120–129.

[Zhen et al., 2017a] Zhen, X., Yu, M., He, X., and Li, S. (2017a). Multi-target regression via robust low-rank learning. IEEE transactions on pattern analysis and machine intelligence, 40(2):497–504.

[Zhen et al., 2016b] Zhen, X., Yu, M., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2016b). Descriptor learning via supervised manifold regularization for multi-output regression. IEEE transactions on neural networks and learning systems, 28(9):2035–2047.

[Zhen et al., 2017b] Zhen, X., Zhang, H., Islam, A., Bhaduri, M., Chan, I., and Li, S. (2017b). Direct and simultaneous estimation of cardiac four chamber volumes by multioutput sparse regression. Medical image analysis, 36:184–196.

[Zheng et al., 2018] Zheng, Q., Delingette, H., Duchateau, N., and Ayache, N. (2018). 3-d consistent and robust segmentation of cardiac images by deep learning with spatial propagation. IEEE transactions on medical imaging, 37(9):2137–2148.

[Zhou et al., 2016] 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 (CVPR).

[Zhou et al., 2011] Zhou, H., Schaefer, G., Celebi, M. E., Lin, F., and Liu, T. (2011). Gradient vector flow with mean shift for skin lesion segmentation. Computerized Medical Imaging and Graphics, 35(2):121–127.

[Zhou et al., 2021] 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.

[Zhou et al., 2018] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. In Deep learning in medical image analysis and multimodal learning for clinical decision support, pages 3–11. Springer.

[Zhu et al., 2019] Zhu, W., Huang, Y., Zeng, L., Chen, X., Liu, Y., Qian, Z., Du, N., Fan, W., and Xie, X. (2019). Anatomynet: deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy. Medical physics, 46(2):576–589.

[Zijdenbos et al., 1994] Zijdenbos, A. P., Dawant, B. M., Margolin, R. A., and Palmer, A. C. (1994). Morphometric analysis of white matter lesions in mr images: method and validation. IEEE transactions on medical imaging, 13(4):716–724.

[Zintgraf et al., 2017] Zintgraf, L. M., Cohen, T. S., Adel, T., and Welling, M. (2017). Visualizing deep neural network decisions: Prediction difference analysis. In 2017 5th International Conference on Learning Representations (ICLR).