Prenatal Assessment of Placenta Accreta Spectrum Disorders from Ultrasound Images Using Deep Learning



Huan Qi

Mansfield College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2019

Acknowledgements

Personal

Firstly I would like to thank my two amazing supervisors. I would like to thank Prof. Alison Noble for her persistent support throughout my DPhil study. I would like to thank Prof. Sally Collins for her constant encouragement and invaluable guidance from the clinical perspective.

I've spent four wonderful years as a member of Mansfield College and Oxford Biomedical Image Analysis Research Group. I would like to thank all my dear friends and colleagues for their support and care. I wish them all the best for the future. Special thanks go to Dr. Yuan Gao, Dr. Yifan Cai, Dr. Ruobing Huang, Dr. Weidi Xie, Dr. Ana Namburete, Dr. Christopher Bridge, Dr. Davis Vigneault, Dr. Mohammad Ali Maraci, Dr. Mohammad Yaqub, Dr. Haiyue Yu, Dr. Hosuk Ryou, Dr. Yipeng Hu, Dr. Xiyu Jiao, Dr. Wei Gan, Dr. Jing Yu, Dr. Guowei Tao, Dr. Hanyang Ye, Dr. Rong Hao, Dr. Lianbo Li, Zixuan Wang, Chunying Zhang, Dr. Guolong Wang, Xue Yao, Shuxiang Cao and Rubi Wu.

Finally, I owe my profound gratitude to my parents Xiaodong Qi and Juyan Wang for their endless love. My heart goes with my beautiful partner Yi Yang, whose love in me is so deep that I would spend a lifetime back-propagating.

Institutional

I am grateful to the China Scholarship Council for funding my DPhil study through a doctoral research award. I would like to thank the China Oxford Scholarship Fund for the support through a Zee award. The support from my *alma mater* Mansfield College in the form of three travel grants and two graduation awards are much appreciated. I would like to thank the following members of the International Society for Abnormally Invasive Placenta for spending their valuable time participating an online evaluation test: Dr. Frederic Chantraine, Dr. Thorsten Braun, Dr. Maddalena Morlando, Dr. Karin Fox, Prof. Pavel Calda, Dr. Vedran Stefanovic, Dr. Louis Marcellin, Dr. Philippe Petit, Prof. Oliver Morel, Dr Amarnath Bhide, Dr. Christos Ioannou, Dr. Katharina v. Weizsaecker, Prof. Boris Tutschek, Dr. Lene Grønbeck. Last but not least, I am grateful to Prof. Aris Papageorghiou and Prof. Yalin Zheng for their feedbacks on this thesis.

Abstract

This thesis addresses the task of developing deep-learning-based tools for assessing placenta accreta spectrum disorders (PASD), a series of placental abnormalities that drastically affect maternal health, on placental ultrasound scans obtained from prenatal examinations.

The contribution of this thesis is a novel design of an image analysis pipeline consisting of computational modules that automatically search for clinically relevant sonographic patterns in placental ultrasound images and perform PASD assessment. This involves identifying informative image planes that clearly depict the anatomy around the placenta within a three-dimensional ultrasound volume; localizing the randomly distributed sonolucent spaces within the placenta (also known as placenta lacunae) caused by abnormal placental vasculature; detecting the utero-placenta interface where abnormal placenta adherence or invasion may occur; integrating ultrasound signs in a collaborative manner to classify PASD. This is the first work to attempt image-based automated prenatal assessment of PASD.

The computational modules are formulated as deep convolutional neural networks, whose parameters are gradually learnt by optimizing objective functions that minimize differences between network outputs and experts' reference annotations. Multi-scale representation learning is considered critical in localizing sonographic patterns with variable sizes and shapes such as patterns associated with placenta lacunae and utero-placental interface. The resulting deep learning models aggregate intermediate features from shallow to deep stages whilst exploiting global contextual cues to model placenta geometry. Apart from architectural refinement, methods are presented to further regularize network behaviours via auxiliary learning, based on additional supervision signals derived from experts' annotations. A complete, fully automated image-based PASD assessment pipeline is presented by effectively combining these computational modules to carry out plane-wise and volume-wise PASD assessment.

A cross-institutional placental ultrasound image analysis database is presented to validate the algorithms proposed in this thesis. The computational modules are compared against a number of competitive benchmark models. Assessment results are further compared to estimates made by a panel of experts who specialize in PASD diagnosis. The results suggest that the proposed PASD assessment pipeline has the potential to be integrated in clinical scenarios to assist in prenatal diagnosis.

Contents

List of Figures **viii**

List of Abbreviations **xiv**

1 Introduction and Preliminaries **1**

1.1 An Overview of the Human Placenta 2

1.2 Placenta Accreta Spectrum Disorders 5

1.2.1 Clinical Definition 5

1.2.2 Prenatal Screening for PASD 6

1.2.3 Limitations of PASD Screening 8

1.3 Aims and Contributions of this Thesis 9

1.3.1 Research Direction 9

1.3.2 Technical Challenges 12

1.3.3 Overview of Contributions 15

1.4 List of Publications 16

2 Literature Review **18**

2.1 Convolutional Neural Network 18

2.1.1 Computational Layer of CNN 20

2.1.2 Loss Function and Optimization 24

2.1.3 Regularization Methods 25

2.1.4 Fully Convolutional Network 26

2.1.5 Classic CNN Architectures for Image Analysis 27

2.2 Deep-learning-based Ultrasound Image Analysis 28

2.2.1 Classification 28

2.2.2 Detection 29

2.2.3 Segmentation 29

2.3 Summary 30

3 Oxford-Hackensack Database of Placenta Accreta Spectrum Disorders 31
3.1 Ultrasound Scanning Protocol . . . 31
3.2 PASD-related Annotations . . . 33
3.2.1 PASD Screen Label . . . 34
3.2.2 Image Information Label . . . 35
3.2.3 Placental Lacunae Dot Annotation . . . 35
3.2.4 Utero-placental Interface Labelling . . . 38
3.2.5 Placenta Previa Screen Label . . . 40
3.3 Relation to Each Chapter . . . 42

4 Weakly Supervised Localization in Placental Ultrasound 44
4.1 Background and Motivation . . . 45
4.2 Weakly Supervised Localization using ResNets . . . 47
4.2.1 Problem formulation . . . 48
4.2.2 Deep Residual Learning . . . 49
4.2.3 Class Activation Mapping for Localization . . . 51
4.3 Experiments & Discussions . . . 53
4.3.1 Dataset . . . 53
4.3.2 Evaluation Metrics . . . 55
4.3.3 Network Variants . . . 56
4.3.4 Weakly-supervised Localization . . . 59
4.3.5 Discussion . . . 59
4.3.6 Limitations . . . 60
4.4 Conclusions . . . 60

5 Placental Lacunae Localization via Multi-scale Layer Aggregation 62
5.1 Background and Motivation . . . 63
5.2 From Dot Annotation to Confidence Map . . . 67
5.3 Deep Layer Aggregation for Representation Learning . . . 70
5.4 Simplified Object Keypoint Similarity . . . 73
5.5 Experiments & Discussions . . . 77
5.5.1 Dataset . . . 78
5.5.2 Implementation Details . . . 79
5.5.3 Performance Analysis . . . 79
5.6 Error Analysis . . . 83
5.6.1 Error Type . . . 84
5.6.2 Sensitivity to Instance Number . . . 90
5.6.3 Limitation . . . 93
5.7 Conclusions . . . 94

6 Utero-placental Interface Detection in Placental Ultrasound 96
6.1 Background and Motivation . . . 97
6.2 UPI-Net: Semantic Edge Detection in Placental Ultrasound . . . 102
6.2.1 Problem Formulation . . . 102
6.2.2 Network Architecture . . . 104
6.3 Experiments and Analysis . . . 109
6.3.1 Dataset . . . 109
6.3.2 Evaluation Protocol . . . 109
6.3.3 Evaluation Metrics . . . 110
6.3.4 Hyperparameter Tuning . . . 111
6.3.5 Implementation Details . . . 112
6.3.6 Results . . . 116
6.3.7 Limitation . . . 117
6.4 Conclusions . . . 118

7 Auxiliary Learning for Utero-placental Interface Detection 119
7.1 Background and Motivation . . . 120
7.2 Exploiting Auxiliary Tasks for Regularization . . . 121
7.2.1 Gaussian Density Map and Adversarial Loss . . . 121
7.2.2 Gradient-sensitive Mask and Spatial Focus . . . 126
7.2.3 UPI-Net++: Architecture and Optimization . . . 132
7.3 Experiments and Analysis . . . 134
7.3.1 Dataset and Evaluation Protocol . . . 134
7.3.2 Implementation Details . . . 135
7.3.3 Performance Analysis . . . 137
7.3.4 Limitations . . . 142
7.4 Conclusions . . . 142

8 Image-based Prenatal Assessment of Placenta Accreta Spectrum Disorders 144
8.1 Background and Motivation . . . 145
8.2 Image-based Prenatal Assessment Pipeline . . . 148
8.2.1 Dataset and Training Protocol . . . 148
8.2.2 Informative Plane Screening . . . 149
8.2.3 Plane-wise Pattern Recognition . . . 155
8.2.4 Collaborative PASD Assessment . . . 156
8.2.5 Experimental Results . . . 159
8.3 Comparison to Human Experts . . . 161
8.4 Conclusions . . . 167

9 Conclusions and Future Work **169**
9.1 Summary of Contributions 169
9.2 Directions for Future Work 173
9.2.1 Three-dimensional Sonographic Pattern Recognition 173
9.2.2 Joint Analysis of B-mode and Doppler Ultrasound 173

Appendices

A Appendix **176**
A.1 Implementation of Algorithms 176

References **178**

List of Figures

1.1 Positions of placenta in the uterus. 3
1.2 Anatomical and vascular structure of the placental circulation system. 4
1.3 Categories of PASD comparing to normal case. 7
1.4 Samples from the Oxford-Hackensack database of PASD demonstrating appearance variation in placental ultrasound scans. 13
1.5 Samples of non-informative image planes in placental ultrasound. . 14

2.1 A classic CNN architecture, LeNet-5 [61], designed for handwriting recognition. 19
2.2 Four commonly used activation functions. 22
2.3 Classic CNN architectures in the task of image classification. 27

3.1 Informative planes (blue) and non-informative planes (red) from two volume samples. Plane indices vary along the axis (marked as green) perpendicular to the sagittal plane. 37
3.2 Dot annotations for PL instances. Image samples with different PASD screen labels are displayed. 38
3.3 Samples with UPI labelling. 39
3.4 Samples with previa screen labels 41

4.1 Three placental ultrasound image samples taken from the Oxford-Hackensack database. Image patches are randomly cropped and illustrated with reference segmentation masks of the follow anatomical structures: placenta (PLA), subcutaneous tissue (ST) and myometrium (MYO). These patches fall into four multi-structure categories: (i) PLA; (ii) PLA+MYO; (iii) PLA+ST; (iv) PLA+MYO+ST. Please note that all segmentation masks that appear here are used solely for illustration purpose rather than training models. 47
4.2 Proposed deep residual network architecture with different sizes of residual units, as listed in Table 4.1. 49

4.3 Proposed pipeline for multi-structure classification. Given an unseen image patch from a placental ultrasound, the network predicts the multi-structure class within. By computing the class activation maps of each class, the network can localize discriminative regions associated with the corresponding anatomical structures. 52
4.4 A custom annotation tool *labelMIL*. 54
4.5 Confusion matrices on the test set for the top three models. 57
4.6 Results of weakly supervised localization with reference segmentation mask of the following anatomical structures: placenta (PLA), subcutaneous tissue (ST) and myometrium (MYO). All images are from the test set. 58

5.1 Illustration of PL centroids (red dots), PL shapes (yellow curves) and false positive PL predictions (crosses) made by deep learning models. Most of the PL-like false positives are gaps formed by the placenta and fetal limbs or skull. 66
5.2 The analysis pipeline that generates a local confidence map around a PL given only a dot annotation. (a) a cropped PL image with the dot annotation in the center, (b) the SLIC over-segmentation of the region, (c) a cluster expansion algorithm yielding a binary mask around the dot annotation, (d) a local confidence map is generated by fitting a Gaussian function at the labelled PL, (e) PL confidence map samples. 68
5.3 Down-sampling networks (DN) with three layer aggregation structures: (a) deep supervision (DN-DS), (b) deeply supervised feature pyramid network (DN-DS-FPN), and (c) deeply supervised iterative deep aggregation (DN-DS-IDA). Intuitively, the box size is proportional to the spatial resolution and the box linewidth to the feature channel number. 71
5.4 U-shape networks (UN) with three layer aggregation structures: (a) deep supervision (UN-DS), (b) deeply supervised feature pyramid network (UN-DS-FPN), and (c) deeply supervised iterative deep aggregation (UN-DS-IDA). Intuitively, the box size is proportional to the spatial resolution and the box linewidth to the feature channel number. 72
5.5 Example images illustrating the use of SOKS to score PL localization. In each image, a red dot denotes the dot annotation of a reference PL and a white cross denotes the prediction. The value next to a pair of dot and cross is the SOKS score between them. Note that every PL prediction has an associated SOKS score to its nearest reference PL. However, there can be unmatched reference PL, for example the one in the bottom-right image. 74

5.6 A simulated example demonstrating the process of computing average precision given a set of detections and reference labels. 76
5.7 Precision-recall curves on the test set of fold-1 at four SOKS levels: $\epsilon \in \{0.3, 0.5, 0.7, 0.9\}$. 81
5.8 Test samples of UN-DS-IDA. The left column presents the input image samples. The middle column presents the network outputs. The right column shows predictions (white cross) and references (red dot), along with the associated SOKS scores for matched pairs. . . . 82
5.9 Example results showing two model outputs, trained using isotropic Gaussian confidence map and the proposed SLIC-based confidence map respectively. 83
5.10 Two main classes of errors are defined based on SOKS scores and the matching scheme introduced in § 5.4. A localization error can be further categorized as a jitter or a miss, while a background error is either a false positive or a false negative. Testing samples are displayed to illustrate errors of different types via color encodings. Best viewed in color. 85
5.11 The frequency of localization errors over all matched PL predictions. 86
5.12 Overall AP improvements obtained after correcting localization errors; evaluated at SOKS levels of {0.3, 0.5, 0.7}. 86
5.13 Change of precision-recall curves before and after correcting *jitter* errors; evaluated at SOKS level of 0.5. 87
5.14 Change of precision-recall curves before and after correcting *miss* errors; evaluated at SOKS level of 0.5. 87
5.15 Overall AP improvements obtained after ignoring background errors; evaluated at SOKS levels of {0.3, 0.5, 0.7}. 88
5.16 Precision-recall curve changed after ignoring BFP errors; evaluated at SOKS level of 0.5. 88
5.17 Precision-recall curve changed after correcting BFN errors; evaluated at SOKS level of 0.5. 89
5.18 Coarse to fine error analysis. The numbers in each plot are the value of $AP_{0.5}$ after sequentially fixing each error type in the following order: *miss, jitter, BFP, BFN*. 90
5.19 Distribution of the number of PL instances per image over the entire dataset. 91
5.20 Sample images with dot annotations from three subsets. 91
5.21 Model sensitivity to instance number, evaluated at SOKS levels of {0.3, 0.5, 0.7}. 92

6.1 Utero-placental interfaces (UPI) are annotated as red curves in placental ultrasound image examples. In PASD cases, the myometrium tends to disappear due to placental invasion, causing weaker contrast around the UPI. Strong placental invasion would produce a bulge-like UPI as shown in (a). In the placenta previa case, UPI usually takes a 'U-shape' over the cervix, as shown in (b). A non-PAS case where UPI separates the placenta from the myometrium is shown in (c), somewhat contaminated by signal dropout. An example with a low-contrast UPI is shown in (d). The indistinct appearance makes detection more challenging. 100
6.2 Semantic edge detection in natural images (examples from SBD [171]) and in placental ultrasound images (examples from the Oxford-Hackensack database of PASD (see Chapter 3). Best viewed in color. 101
6.3 Three multi-scale feature aggregation architectures: (a) HED [140]; (b) CASENet [174]; (c) DS-FPN [116]. 104
6.4 The proposed UPI-Net with a VGG-16 backbone. 106
6.5 A global context (GC) block, which takes a $B \times C \times H \times W$ feature map and produces a refined feature map of the same size. 107
6.6 A convolutional group-wise enhancement (CGE) block, which takes a $B \times C \times H \times W$ feature map and produces a refined feature map of the same size. 108
6.7 A diagrammatic demonstration of the nested cross-validation evaluation protocol. 110
6.8 Hyper-parameter searching for UPI-Net, where an iterative strategy is applied for better efficiency. 112
6.9 Fold-wise performance comparison among UPI detectors. 113
6.10 Predictions from the proposed UPI-Net model and other benchmarks. UPI-Net suppresses a number of UPI-like false positives compared with other methods. 114
6.11 Semantic entities learnt by UPI-Net. 115
6.12 Sub-optimal predictions made by UPI-Net and HED indicates there is room to improve performance. 117

7.1 Constructing auxiliary supervision signals given an expert annotation. 122
7.2 Side-outputs from HED [140]. 123
7.3 Loss comparison. 123
7.4 An illustration of the process to generate Gaussian density maps. The hyper-parameter σ can be tuned to obtain maps of different scales. 124

7.5 An illustration of training a conditional generative adversarial network (cGAN) to produce UPI predictions that look like Gaussian density maps. The discriminator Φ learns to distinguish fake inputs (predictions from the generator conditioned on input images) from real inputs (Gaussian density maps conditioned on input images), while the generator Ψ learns to fool the discriminator. 125
7.6 UPI-Net++. 128
7.7 Gradient-sensitive Masking. 130
7.8 Supervision augmentation samples. 130
7.9 Masking attention block. 131
7.10 Multi-scale global context (MGC) block. 132
7.11 A diagram of the 6-fold nested cross-validation evaluation protocol. 134
7.12 Representations before (middle column) and after (right column) fed into a MGC block. Samples from different scales are displayed. . . . 138
7.13 Effects of L_{adv} and L_{mask} on UPI detections. (a) L_{adv} enforces spatial continuity. Red dotted boxes highlight some failure cases when the model is trained without L_{adv}. (b) L_{mask} promotes gradient-sensitive spatial attention, which helps filter out unwanted false positives, as marked by the red dotted boxes. 139
7.14 Predictions from the proposed UPI-Net++ and other benchmark models. As with Figure 6.10, red curves denote the reference UPI and green curves denote UPI predictions. 141

8.1 An image-based prenatal assessment pipeline for PASD, with three key stages enclosing four computational modules. 147
8.2 Data distribution and training/test protocol. (a) PASD screen label distribution with respect to placental ultrasound volumes (planes); (b) placenta previa screen label distribution with respect to placental ultrasound volumes (planes); (c-i) the dataset is first divided into training splits, validation splits and test splits; (c-ii) deep-learning models are trained on training splits and tuned on validation splits; (c-iii) trained models from each stage are then evaluated on test splits. As with § 7.3.1, the same 6-fold nested cross-validation are adopted to compensate for the lack of test data. 149
8.3 Stage 1 workflow. Raw placental ultrasound volumes are sliced along sagittal planes and classified by the informative plane screening network. An illustrative sample is presented inside the dotted box, where the plane-wise probabilities are plotted by varying plane indices. 151

8.4 An illustrative example of applying the *informative plane screening network* on the sagittal planes of a sample volume. In the right figure, planes with white backgrounds are filtered out by the model and the remaining ones are selected for subsequent analysis. 151
8.5 ROC curves for informative plane screening networks. The blue curves denote models trained via transfer learning while the red curves denote those trained from scratch. 152
8.6 Stage 2 workflow. Informative planes are fed into two pattern recognition branches for accurate PL localization and UPI detection respectively. 154
8.7 UPI post-processing module in Stage 2-2. It takes UPI heatmaps from UPI-Net++ as inputs and performs anatomically grounded extrapolation to generate UPI masks that serve as spatial attention maps for subsequent analysis. 154
8.8 Stage 3 workflow. The proposed PAN takes as input a tuple of information: an informative plane, its estimated PL heatmap from Stage 2-1 and its extrapolated UPI mask from Stage 2-2. It then outputs PASD classification scores (primary task) and placenta previa classification scores (auxiliary task) in a plane-wise manner. Volume-wise scores are calculated via plane aggregation. 157
8.9 Architecture of PAN. 157
8.10 (a) A mask attention block to enforce UPI spatial attention; (b) Convolutional layers inside PAN's blocks. 158
8.11 CAM samples with PASD screen labels (white), generated from PAN-6 and baseline respectively. 161
8.12 Plane-wise ROC curves for Fold 1 and Fold 2. 163
8.13 Plane-wise ROC curves for Fold 3 and Fold 4. 164
8.14 Plane-wise ROC curves for Fold 5 and Fold 6. 165

List of Abbreviations

2D, 3D . . . . Two-dimensional, three-dimensional.

PASD . . . . . Placenta accreta spectrum disorders.

PL . . . . . Placenta lacunae

UPI . . . . . Utero-placental interface

PLA . . . . . Placenta *in multi-structure categories*

ST . . . . . Subcutaneous tissue *in multi-structure categories*

MYO . . . . . Myometrium *in multi-structure categories*

DN . . . . . Down-sampling network

UN . . . . . U-shape network

GC . . . . . Global context

CGE . . . . . Convolutional group-wise enhancement

MGC . . . . . Multi-scale global context

FIGO . . . . . International Federation of Gynecology and Obstetrics

UPC . . . . . Utero-placental circulation

FPC . . . . . Feto-placental circulation

CS . . . . . Caesarean section

FGR . . . . . Fetal growth restriction

BMI . . . . . Body mass index

CNN . . . . . Convolutional neural network

RGB . . . . . Red-green-blue *color channels of an image*

FCN . . . . . Fully convolutional network

ROC . . . . . Receiver operating characteristic *curve*

AUC . . . . . Area under ROC curve

ITK . . . . . Insight Toolkit

VTK . . . . . The Visualization Toolkit

NP . . . . . Normal placenta

AAP . . . . . Abnormally adherent placenta

AIP . . . . . Abnormally invasive placenta

CAM . . . . . Class activation mapping

GMP . . . . . Global max pooling

DS . . . . . Deep supervision (DS)

FPN . . . . . Feature pyramid network

IDA . . . . . Iterative deep aggregation

TP . . . . . True positive

FP . . . . . False positive

FN . . . . . False negative

IG . . . . . Ignored *prediction*

AP . . . . . Average precision

BFP . . . . . Background false positive

BFN . . . . . Background false negative

NMS . . . . . Non maximum suppresion

SOKS . . . . . Simplified object keypoint similarity

HED . . . . . Holistically-nested edge detector

SBD . . . . . Semantic boundaries dataset

PAN . . . . . PASD assessment network

1

Introduction and Preliminaries

Contents

1.1 An Overview of the Human Placenta 2
1.2 Placenta Accreta Spectrum Disorders 5
1.2.1 Clinical Definition 5
1.2.2 Prenatal Screening for PASD 6
1.2.3 Limitations of PASD Screening 8
1.3 Aims and Contributions of this Thesis 9
1.3.1 Research Direction 9
1.3.2 Technical Challenges 12
1.3.3 Overview of Contributions 15
1.4 List of Publications 16

This thesis develops automated image analysis algorithms to perform prenatal assessment of placenta accreta spectrum disorders (PASD) on ultrasound scans. This chapter starts with a basic overview of the human placenta, followed by descriptions of PASD and the associated prenatal ultrasound screening guidelines used in clinical practice.

1.1 An Overview of the Human Placenta

The human placenta[1] is a feto-maternal hemochorial villious organ attaching to the uterine wall during pregnancy. Functionally, the placenta hosts nutrient and gas exchange between the fetus and the mother while also plays an important immunomodulatory role [1]. As an important endocrine organ, it produces hormones and growth factors to regulate the course of pregnancy, support and promote fetal growth, and initiate parturition [2].

Abnormal placentation affects maternal health, as shown by the development of gestational hypertension and pre-eclampsia [3]. Placental maldevelopment as well as dysfunction affects the growth of the fetus, which may lead to prematurity, fetal growth and neurodevelopment abnormalities [4, 5]. Studies have shown that placental health also impacts on the lifelong health of both the mother and the fetus [6]. A shallow invasion of spiral arteries in placentation may impair development of the human autonomic nervous system, causing sudden cardiac death later in life in the worst case [7]. Barker and Thornburg showed that variations in gross placental morphology at birth may predict certain disorders in later life [8]. Therefore, investigation of placental structure, development and function is of great significance to improve understanding of how the placenta affects human health. Preventive and therapeutic interventions have potential to achieve lifelong impact [9]. How to optimize detection, assessment and treatment pipelines for placental abnormalities remain open questions at the intersection of life science, medicine, and engineering.

A basic overview of placental anatomy and vasculature is presented here, which is relevant to the development of placenta accreta spectrum disorders (PASD). For a detailed discussion, the reader could consider a recent book edited by Kay, Nelson and Wang [10], which examines recent research into placental function from a clinical perspective, and another one by Wang and Zhao [2] on vascular biology of the placenta [2].

Figure 1.1 shows three schematic stereotypes of placental position inside the uterus. Note that the distance between the lower placental edge and the internal

[1]Hereafter, it shall be referred to as *placenta* for simplicity.

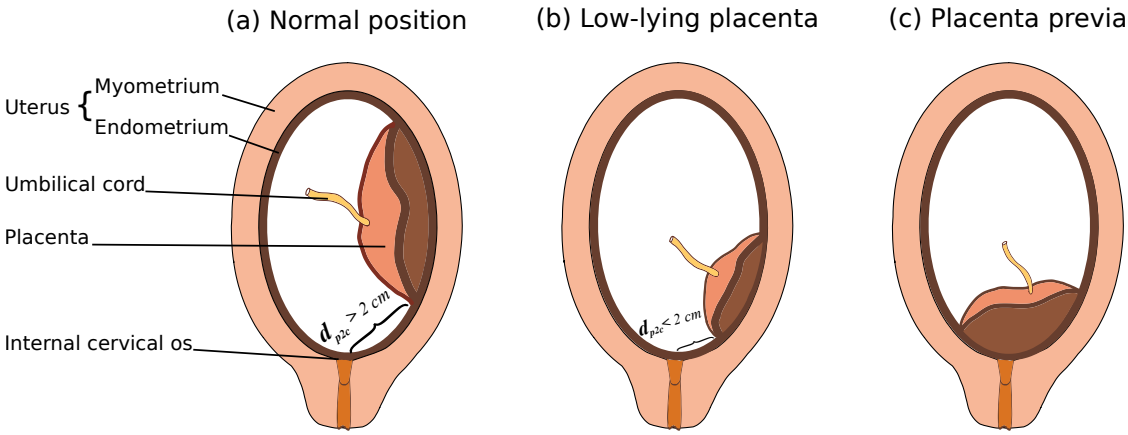


Figure 1.1: Positions of placenta in the uterus.

Type	Delivery	Description
Normal placenta	Vaginal delivery	The lower placenta edge is more than 2 cm away from the internal cervical os.
Low-lying placenta	Caesarean delivery	The lower placenta edge is less than 2 cm away from the internal cervical os.
Placenta previa	Caesarean delivery	The placenta partially or completely covers the internal cervical os.

Table 1.1: Types of placenta position, and the corresponding delivery mode and abnormality description.

cervical os, denoted as d_{p2c}, can provide clinical guidance on the safe choice of delivery mode [11–13]. This thesis follows the placenta previa categorization system introduced in the green-top guideline No. 27a by the Royal College of Obstetricians and Gynaecologists [14]. A normally positioned placenta (Figure 1.1a) has a d_{p2c} greater than 2 cm. On the contrary, a low-lying placenta lies close to the internal cervical os with a d_{p2c} less than 2 cm, but does not cover the os (Figure 1.1b). Finally, a placenta previa (Figure 1.1c) refers to a placenta that overlies the internal cervical os [15], completely or partially lying within the lower uterine segment, as listed in Table 1.1. It is suggested that women with placenta previa should undergo an additional transvaginal sonogram in the late third trimester to measure d_{p2c}. A low-lying placenta or a placenta previa requires delivery by cesarean section [15]. It is reported that placenta previa complicates approximately 0.3%-0.5% of pregnancies

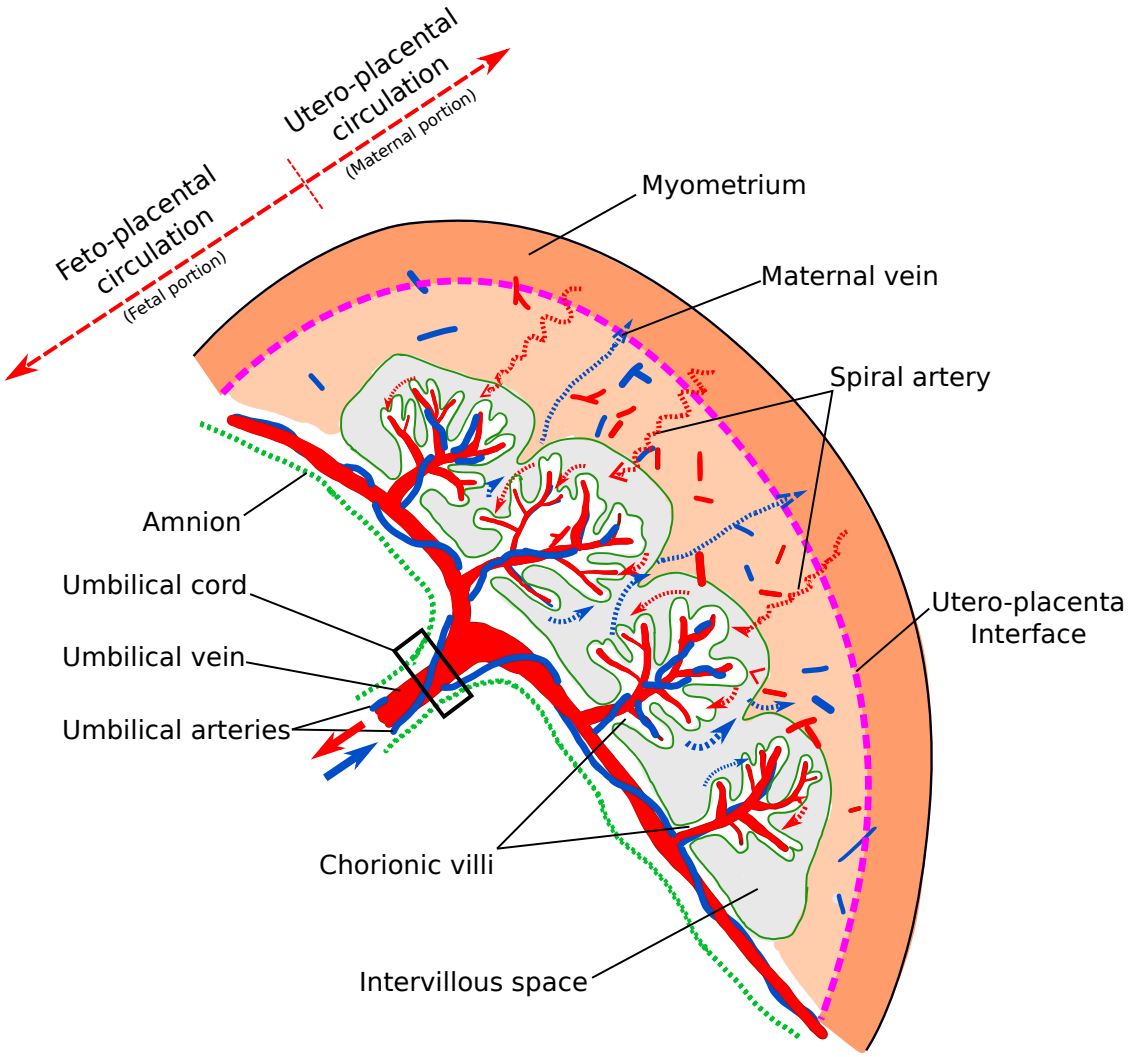


Figure 1.2: Anatomical and vascular structure of the placental circulation system.

[16]. With the progression of pregnancy, more than 90% of the low-lying placenta cases identified in the first / second trimester will become normal cases due to the phenomena known as *placental migration*, where the placenta grows away from the lower uterine segment for a better vascular environment and the lower uterine segment develops thereby appearing to move the placenta upwards away from the os [17, 18]. Assessing placenta previa is a relevant topic in this thesis as it is reported to be one of the major risk factors of PASD in a recent study by Thurn *et al.* [19].

A graphical demonstration of placental anatomy and vasculature is displayed in Figure 1.2. This depicts a fully functional placental circulation, which is usually established by the end of the first trimester. In order to coordinate blood supplies

from both the maternal side and the fetal side, the placenta develops two separate blood circulatory systems: the utero-placental circulation (UPC) and the feto-placental circulation (FPC) [2]. UPC starts when the oxygenated and nutrient-rich maternal blood flow (from the maternal circulation) into the intervillous space through spiral arteries. The exchange of oxygen and nutrient occurs as the maternal blood flows around the terminal villi in the intervillous space. The in-flowing effect pushes deoxygenated blood into maternal veins and back to the maternal circulation. Meanwhile, FPC uses the umbilical veins to carry oxygen and nutrient to the fetal systemic circulation. The umbilical arteries carry deoxygenated and nutrient-depleted fetal blood back to the chorionic villi [2]. Note that there is *no intermingling* of maternal and fetal blood in the placenta. During pregnancy, UPC and FPC gradually shape the anatomy of the maternal portion and fetal portion of the placenta. In the presence of vascular abnormalities, structural abnormalities are often observed as well, allowing sonographers (and image analysis algorithms) to deduce the internal vasculature from B-mode ultrasound images. Placental function depends on its vascular development [2]. Different levels of placental vascular lesions have been confirmed in pregnancies complicated by pre-eclampsia and fetal growth restriction (FGR) [20]. It is also reported that certain associations exist between placental vascular lesions and medically indicated or spontaneous preterm delivery [21].

1.2 Placenta Accreta Spectrum Disorders

1.2.1 Clinical Definition

Placenta accreta spectrum disorders (PASD) are a series of placental abnormalities that cover conditions where the placenta abnormally adheres to the uterus deeply or even invades into it. Histopathologically, there are three types of PASD (Figure 1.3): (i) placenta creta[2], where the chorionic villi attach to the myometrium; (ii) placenta

[2]Placenta creta is also known as placenta accreta. The terminology of this thesis follows the latest International Federation of Gynecology and Obstetrics (FIGO) consensus guidelines by Jauniaux et al [22].

increta, where the chorionic villi invade the myometrium; (iii) placenta percreta, where the chorionic villi penetrate the myometrium and the uterine serosa and sometimes adjacent pelvic organs [23]. In the absence of proper prenatal assessment, forced removal of a placenta with PASD can lead to massive and potentially life-threatening intrapartum and postpartum hemorrhage [24, 25]. PASD have also gradually become a leading cause of emergency hysterectomy (a surgical operation to remove all or part of the uterus), accounting for more than 50% of all emergency hysterectomies [26, 27].

Maternal mortality for the most severe end of the PASD (percreta) has been quoted to be as high as 7% in a study published in 1996 [28]. The major risk factors for PASD are prior caesarean section (CS), other uterine surgery, assisted reproduction techniques and placenta previa [19, 29–31]. The single greatest risk is CS with the risk of PASD increasing seven-fold after one prior CS, and 56-fold after three or more CS [19]. In 1970, the CS rate in the USA was 5.5% [32], comparing to 24.7% in 1988 and 32.2% in 2014, the latest statistics. The incidence of PASD has increased from 1:25,000 in the 1950s, to rates of 1: 2,500 in the 1980s, paralleling the rise in CS rates [31, 33, 34]. Recently, it was reported that PASD rates are as high as 1:533 in the USA and 1:588 in Canada [30, 35]. It is estimated that by 2020 the USA will have a 56% CS rate, accounting for an additional 4504 cases of PASD and 130 maternal deaths annually [36].

Such drastic change in PASD statistics sees the urgency to improve corresponding diagnostic approaches as well as obstetric management. Undiagnosed PASD cases may lead to significantly more emergency hysterectomies and mass transfusions during or immediately after delivery [37]. Early and reliable prenatal diagnosis of PASD is desperately needed to reduce maternal morbidity and mortality.

1.2.2 Prenatal Screening for PASD

The standard method for prenatal screening for PASD is via an ultrasound scan [38–42]. Sonographers with relevant expertise assess placental ultrasound scans with subjective interpretation of representative sonographic findings or signs in

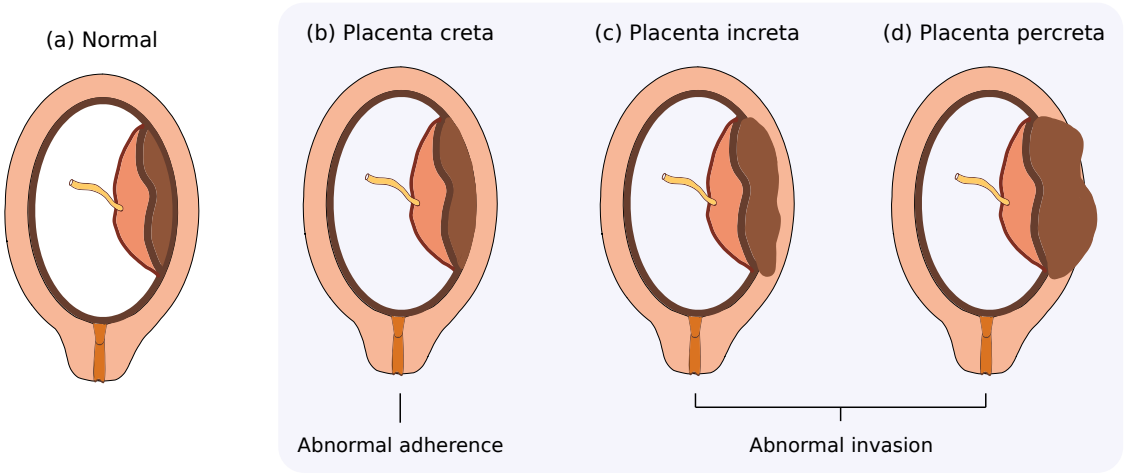


Figure 1.3: Categories of PASD comparing to normal case.

two-dimensional (2D) B-mode and color Doppler ultrasound images. Many signs and patterns have been extensively studied and verified by clinical research. To improve consistency in the definition, terminology, and diagnostic criteria, the International Federation of Gynecology and Obstetrics (FIGO) released consensus guidelines on PASD in 2018 [22], aiming to standardize the descriptions and minimum requirements for prenatal ultrasound screening. Specifically, a series of unified descriptors for sonographic patterns in PASD were proposed by the European Working Group on Abnormally Invasive Placenta (EW-AIP) [43]. Table 1.2 lists descriptors for 2D B-mode ultrasound, which is the relevant imaging modality in this thesis. For guidelines on color Doppler ultrasound, please refer to Collins et al [43] and Jauniaux et al [22].

It is worth noting that almost all sonographic patterns in Table 1.2 describe some abnormalities near / on the *utero-placental interface* (UPI). The only exception is the pattern related to abnormal placental lacunae (PL). UPI is the anatomical interface that separates the placenta from the uterus. Loss of the clear zone, bladder wall interruption, and myometrial thinning describe visual scenarios located at the UPI, while placental bulge and focal exophytic mass can be directly reflected by the geometric distortion of the UPI. It was reported by Collins et al [40] that women at risk of PASD developed abnormal vasculature near the UPI or even further to myometrium and uterine serosal surface. Another study by Chantraine

Descriptors	Sonographic patterns
Loss of the *clear zone*	Loss or irregularity of the hypoechoic plane in the myometrium underneath the placental bed (the *clear zone*)
Bladder wall interruption	Loss or interruption of the bright bladder wall (the hyperechoic band or "line" between the uterine serosa and the bladder lumen)
Myometrial thinning	Thinning of the myometrium overlying the placenta to <1 mm or undetectable
Placental bulge	Deviation of the uterine serosa away from the expected plane, caused by an abnormal bulge of placental tissue into a neighboring organ, typically the bladder. The uterine serosa appears intact but the outline shape is distorted
Focal exophytic mass	Placental tissue seen breaking through the uterine serosa and extending beyond it. Most often seen inside a filled urinary bladder
Abnormal placental lacunae	Presence of numerous lacunae including some that are large and irregular often containing turbulent flow visible in grey-scale imaging

Table 1.2: Unified descriptors for PASD-related sonographic patterns in B-mode ultrasound images.

et al [44] suggested that the arrival of high-velocity maternal blood from radial or arcuate arteries may promote the development of PL [45]. Therefore, this thesis argues that the incidence of these PASD-related sonographic patterns is attributed to the abnormal vasculature development of the placenta complicated by PASD during pregnancy, and the UPI and PL are two key *regions of interest (ROI)* for prenatal screening for PASD.

1.2.3 Limitations of PASD Screening

A number of factors limit the success of prenatal screening for PASD, as discussed in [22]. First and foremost, many of these factors are a consequence of the fact that prenatal diagnosis of PASD remains *subjective*. The accuracy depends on the experience of the operator, which has been itself limited by many factors, such as lack of training and rarity of the condition. Specifically, the presence or absence of the PASD-related descriptors in Table 1.2 is influenced by the operator's interpretation of what constitutes that descriptor, which is particularly challenging in ultrasound images due to the indistinct appearance of visual features (e.g. boundaries, blobs),

variations among subjects of different body habitus, and the presence of artefacts (e.g. speckle, acoustic shadow, signal dropout). For instance, the effect of signal attenuation is associated with the body mass index (BMI) of subjects [46]; and some dropout in placental ultrasound imaging can be attributed to absorption effects caused by scars due to previous caesarean delivery surgeries. Therefore, interpretation of PASD-related descriptors is subjective and lacking in reproducibility [40], which is inclined to misdiagnosis if the operator has minimal experience with placental ultrasound or PASD. Studies have shown that the sensitivity and specificity of screening placenta previa and PASD using B-mode ultrasound are consistently higher when performed by experts than by non-experts [41, 47].

The variation of scanning conditions and ultrasound machine settings may also influence prenatal diagnosis of PASD. For instance, ultrasound examinations usually require women to have full bladders (200-300 mL) because the bladder outline helps with identification of the lower uterine segment, making assessment of the placental position, evaluating bladder wall interruption and placenta bulge more accessible. Failing to get a full bladder can result in a lower imaging quality near the UPI, thus affecting PASD screening. Moreover, appropriate machine settings can help reduce ultrasound artefacts for the process [48]. Excessive probe pressures may mislead the interpretation of *loss of the clear zone* in Table 1.2. In practice, appropriate finetuning of these factors contribute to the accuracy of PASD prenatal diagnosis.

Currently, PASD screening is not routinely performed in the mid-trimester ultrasound scan for detailed fetal anatomy examination. Training programs similar to those existing for the screening of fetal anatomical defects, such as congenital heart defects [22], need establishing such that sonographers understand the risk of PASD and are aware of its referral pathway for further examination if they have any concern.

1.3 Aims and Contributions of this Thesis

1.3.1 Research Direction

Upon reviewing the descriptors for PASD-related sonographic patterns (Table 1.2), it is natural to speculate that training of sonographers would require showcasing

placental ultrasound image samples demonstrating these patterns. Particularly, learning to relate placenta anatomy (e.g. from textbooks) to specific sonographic patterns by using *global and local contextual cues* needs a lot of practice on representative samples. Moreover, the ability to robustly identify relevant semantics irrespective of potential variations described in Section 1.2.3 is vital in image-based decision-making. For instance, in order to decide whether there is a *loss of clear zone* in a given placental ultrasound image, a sonographer needs to first know the approximate location of the placenta and the myometrium, then accurately localize the boundary that separates them (i.e. the UPI) and identify the clear zone within. Similarly, a robust tracking of the placental boundary is important to determine the existence of a *placental bulge*.

As discussed in Section 1.2.3, it is clear that improving sonographers' expertise via effective training programs has an important part to play in improving prenatal screening accuracy for PASD. Interestingly, with the rapid advances in the areas of computer vision and machine learning, some of the requirements and goals listed above are actually achievable through carefully formulated computational models. It is therefore natural to ask whether a new direction can be explored to improve prenatal screening accuracy for PASD, namely by developing image analysis algorithms to reduce the level of skills required for accurate diagnosis using placental ultrasound scans and reduce the influence of subjective errors on important decision-making steps.

A number of works have successfully applied machine learning (especially deep learning) algorithms to a variety of medical applications involving ultrasound images (see Section 2.2 for a review of the literature). However much of this work only concentrates on some intermediate steps (such as ROI segmentation, ROI detection) without devising an effective image analysis pipeline. In this thesis, the aim is to develop such a pipeline to assist sonographers in performing effective prenatal screening for PASD. Previous work in automating placental abnormality detection has been minimal, and particularly there is no relevant work that has covered PASD screening from the perspective of image analysis and machine learning.

An effective diagnostic pipeline may consist of many computational modules designed for specific goals. Conceptually, there are many feasible goals that, if achieved, can optimize the current prenatal ultrasound screening process for PASD, such as the following:

- For 3D ultrasonography, each plane along each axis can be viewed as a 2D ultrasound image. Automated algorithms can tag these planes with predefined attributes / labels (e.g. what anatomy is visible), as a means to obtain semantic-level statistics and indices for review purpose.

- During ultrasound scans, automatically segmented ROIs (e.g. thresholded binary maps or continuous probabilistic heatmaps) could be visually highlighted on the screen to help the less experienced sonographers better interpret the images. Moreover, subsequent statistical analysis built upon automatic segmentation techniques may help reveal hidden correlations that lead to discovery of effective biomarkers related to diagnosis.

- Recognized sonographic landmarks / patterns can be used to guide the movement of the probe to reach the relevant plane of interest.

- Through machine learning, it is possible for algorithms to discover robust and discriminative visual descriptors, not listed in the current clinical guidelines (Table 1.2), that contribute to the prenatal screening. Obviously further studies need to be carried out to verify the effectiveness of such descriptors.

- There are usually several ultrasound scans throughout the pregnancy. As a longitudinal study, automated algorithms could integrate multiple scans of the same person to monitor the development of relevant anatomical structures and biomarkers. Ideally such a study could help obtain more relevant diagnostic information.

Developing algorithms to achieve all these listed goals would extend far beyond the scope of this thesis. However there are fundamental questions that must be properly solved in order to develop automated diagnostic pipelines. Discovering and

developing a minimal number of necessary computational modules to constitute an effective prenatal screening pipeline for PASD is the subject of this thesis, with the hope that this work would act as an experimental feasibility analysis and inspire further work to explore this research direction.

1.3.2 Technical Challenges

There are a number of technical challenges associated with the design of an automated prenatal screening pipeline. Broadly, they can be divided into two categories in terms of relevance to the disease / abnormality itself, namely PASD-related challenges (i.e. those encountered only in PASD screening process) and general challenges (i.e. those most likely shared among many medical applications).

Figure 1.4 shows some image samples from the Oxford-Hackensack database of PASD (see Chapter 3 for more details) and illustrates some PASD-related challenges. For instance, PL are sonolucent spaces within the placenta that appear to be randomly distributed with irregular shapes and have unpredictable size and number (Figure 1.4a-b). It has been observed that numerous, large, and irregular PL are more likely to occur in PASD cases (Figure 1.4a) than in normal cases [43], due to the abnormal vasculature development of the placenta complicated by PASD (see Section 1.2.2). However, it may be difficult for automated algorithms to (1) accurately localize PL, and (2) quantify such visual difference between PASD and normal cases. Due to the indistinct appearance of anatomy in placental ultrasound images, usually compounded by variable contrast and artefacts (Figure 1.4g), a number of PL-like blobs (i.e. false positives) may appear to confuse the algorithms to make false localization. Moreover, the descriptor *abnormal placental lacunae* in Table 1.2 is not quantifiable as there is not a definitive decision boundary. Similar challenges may occur for UPI detection, involving unquantifiable descriptors such as *myometrial thinning, bladder wall interruption, and placental bulge* (Figure 1.4c-g).

There are some general technical challenges shared by image-based medical applications. First, inter-site variability in image properties can pose a great challenge to image analysis algorithms when they are applied on data from *unseen*

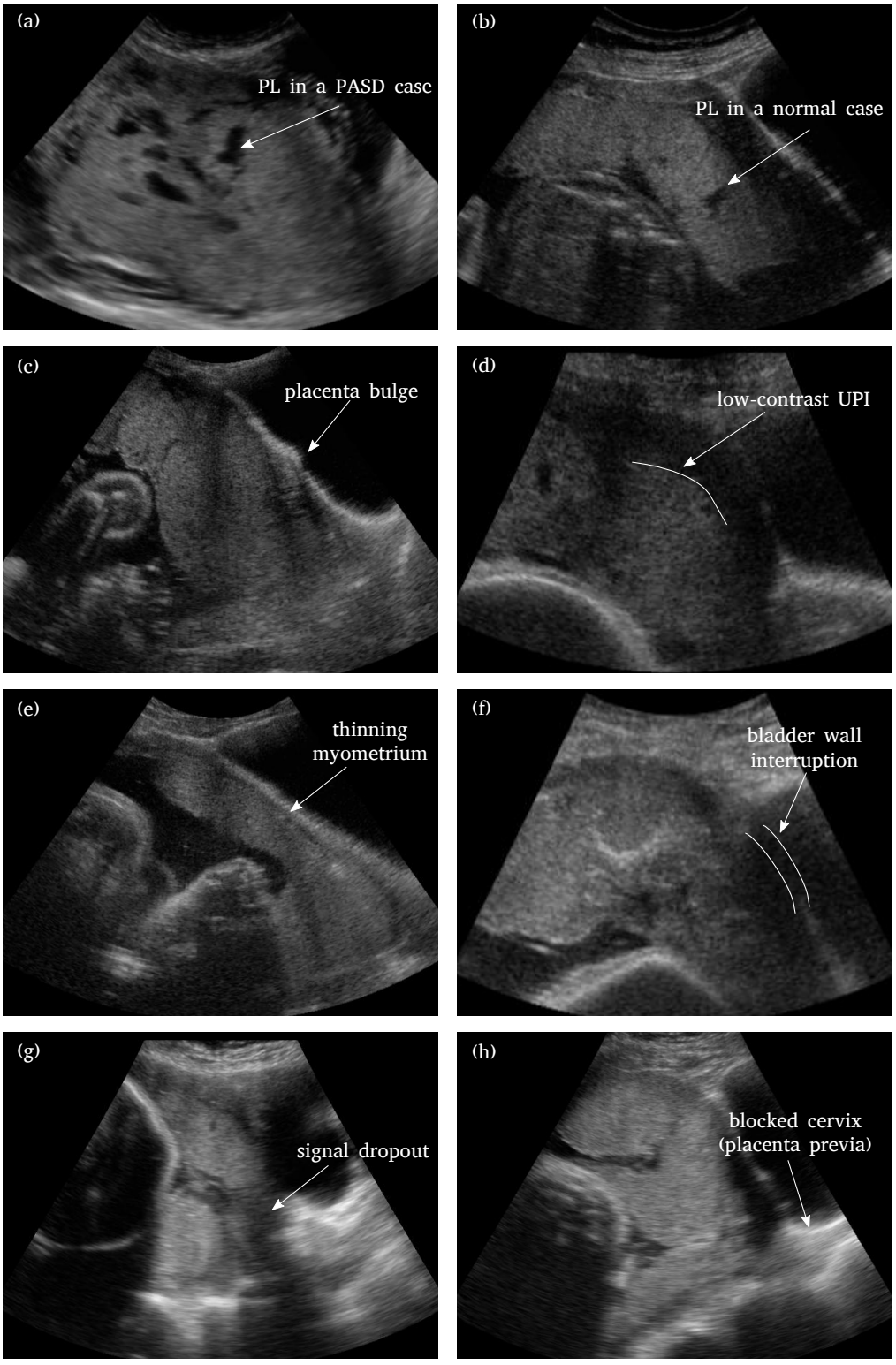


Figure 1.4: Samples from the Oxford-Hackensack database of PASD demonstrating appearance variation in placental ultrasound scans.

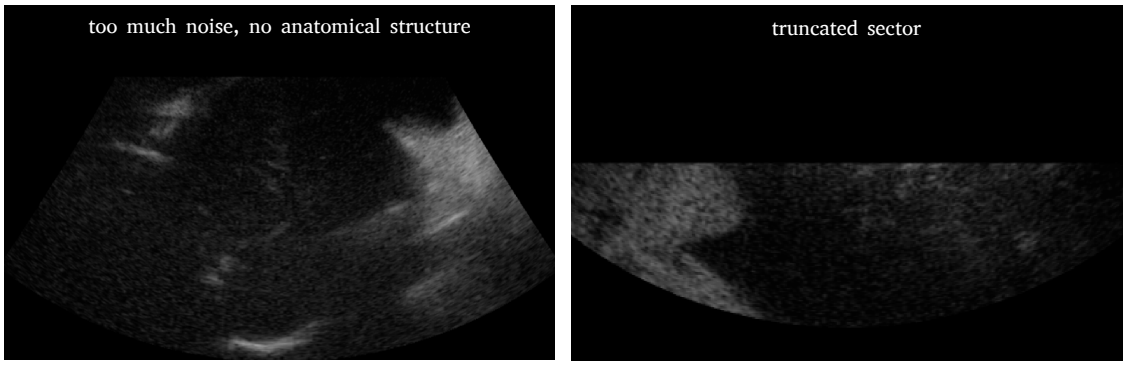


Figure 1.5: Samples of non-informative image planes in placental ultrasound.

sites or machines not included in the available data pool [49]. This results from the fact that many algorithms assume that all the data processed and to process in the future should generally come from a common data distribution. Second, when working on 3D raw data collected directly from clinical sessions there are often a number of image planes that are not relevant to the on-going task. This may include image planes that are largely truncated or are contaminated too much by noise to be interpretable. Take placental ultrasound as an example. Figure 1.5 contains two samples from a 3D scan. Neither of them includes any information relevant for prenatal screening for PASD. In practice a pre-processing module that determines whether an image plane is informative or not should be incorporated into the pipeline. Third, verifying the effectiveness of the proposed automated diagnosis tool is necessary, which often leads to comparing against other solutions. A solution may be some alternative tool from previous work, or even diagnosis made directly by human experts, which can be challenging in practice as it often involves investing extra efforts and resources. Additionally, computational efficiency can also be a significant challenge if algorithms are designed to be deployed in a real-time fashion to interact with the operator.

This thesis aims to develop an automated prenatal screening pipeline for PASD in the face of these challenges. Attempts to overcome many of these represent proofs of concept, with the hope that they would provide some insight for future work in a similar clinical scenario.

1.3.3 Overview of Contributions

This thesis proposes an algorithmic pipeline that considers the task of assessing PASD using placental ultrasound scans (Chapter 8). This pipeline consists of computational modules designed for specific goals. A weakly supervised machine learning model is used to classify and localize anatomical structures in an unseen placental ultrasound image (Chapter 4). Moreover, predictions are made to localize PL using a deep convolutional neural network (Chapter 5). Integration of global contextual cues enables another deep-learning model to accurately detect UPI (Chapter 6). Furthermore, a novel architecture is proposed to improve UPI detection via auxiliary learning (Chapter 7). All the experiments, evaluations, and verification processes are performed on a new database, namely the Oxford-Hackensack database of PASD (Chapter 3). Here are the contributions of this thesis:

1. The placental ultrasound scans used in this thesis were originally collected in an obstetrics research project [40, 50, 51]. The major contribution the author made in Chapter 3 is to provide a variety of PASD-related manual annotations to complement the original PASD screen label from histopathological examinations, thus constitute the Oxford-Hackensack database of PASD. It serves as a reference database to train and test the proposed machine learning algorithms and perform a variety of quantitative evaluations.

2. The author proposes a novel algorithm to classify and localize anatomical structures around the placenta using only image-level labels in Chapter 4. This weakly-supervised classification model is built as a deep convolutional neural network while the localization is achieved by computing class activate maps.

3. In Chapter 5, the author proposes a novel multi-scale aggregation architecture to localize PL within a placental ultrasound image. To carry out performance evaluation, a simplified object keypoint similarity score is introduced. Experimental results show that the proposed PL detector outperforms several widely used feature aggregation architectures.

4. UPI detection is investigated as a key module in the proposed screening pipeline. In Chapter 6, the author proposes a deep learning architecture capable of modelling global contextual cues in a lightweight fashion, which outperforms several competitive benchmarks as shown in the experiments.

5. In Chapter 7, the author further proposes a novel auxiliary-learning-based model to improve UPI detection performance by constructing two auxiliary tasks that further regularize network behaviours without requiring extra manual annotation.

6. In Chapter 8, the author proposes an automated prenatal screening pipeline for PASD by integrating computational modules in previous chapters. Experimental results suggest that the proposed pipeline matches the performance of experts at identifying normal placenta and abnormally adherent placenta while outperforms experts in identifying abnormally invasive placenta.

1.4 List of Publications

This thesis is partly based on the following published work:

- Huan Qi, Sally Collins and Alison Noble, *Knowledge-guided Pretext Learning for Utero-placental Interface Detection*, In Proceedings of the 23^{rd} International Conference on Medical Image Computing & Computer Assisted Intervention (MICCAI), 2020

- Huan Qi, Sally Collins and Alison Noble, *UPI-Net: Semantic Contour Detection in Placental Ultrasound*, In Proceedings of the International Conference on Computer Vision (ICCV) Workshop for Visual Recognition for Medical Images (VRMI), 2019 (oral)

- Huan Qi, Sally Collins and Alison Noble, *Automatic Lacunae Localization in Placental Ultrasound Images via Layer Aggregation*, In Proceedings of the 21^{st} International Conference on Medical Image Computing & Computer Assisted Intervention (MICCAI), 2018 (poster)

- Huan Qi, Sally Collins and Alison Noble, *Weakly Supervised Learning of Placental Ultrasound Images with Residual Networks*, In Proceedings of Medical Image Understanding and Analysis (MIUA), 2017 (oral)

This thesis is partly based on the following manuscripts:

- Huan Qi, Sally Collins and Alison Noble, *Three-dimensional Placental Lacunae Localization using Deep Multi-scale Feature Aggregation*, 2020
- Huan Qi, Sally Collins and Alison Noble, *Deep Auxiliary Learning for Utero-placental Interface Detection in Placental Ultrasound*, 2020
- Huan Qi, Sally Collins and Alison Noble, *Knowledge-augmented Deep Learning for Prenatal Assessment of Rare Placental Disorders*, 2020

2
Literature Review

Contents

2.1 Convolutional Neural Network 18
2.1.1 Computational Layer of CNN 20
2.1.2 Loss Function and Optimization 24
2.1.3 Regularization Methods 25
2.1.4 Fully Convolutional Network 26
2.1.5 Classic CNN Architectures for Image Analysis 27
2.2 Deep-learning-based Ultrasound Image Analysis 28
2.2.1 Classification 28
2.2.2 Detection 29
2.2.3 Segmentation 29
2.3 Summary 30

This chapter reviews the relevant literature to this thesis. It starts with an overview of the convolutional neural network (CNN), a class of deep-learning models that is most commonly applied to image analysis, with respect to architecture, optimization and recent advances in visual recognition tasks (§ 2.1). Deep-learning-based ultrasound image analysis studies are briefly reviewed in § 2.2.

2.1 Convolutional Neural Network

This section briefly introduces the basics of the commonly used computational layers in CNNs and some representative architectures that have recently achieved

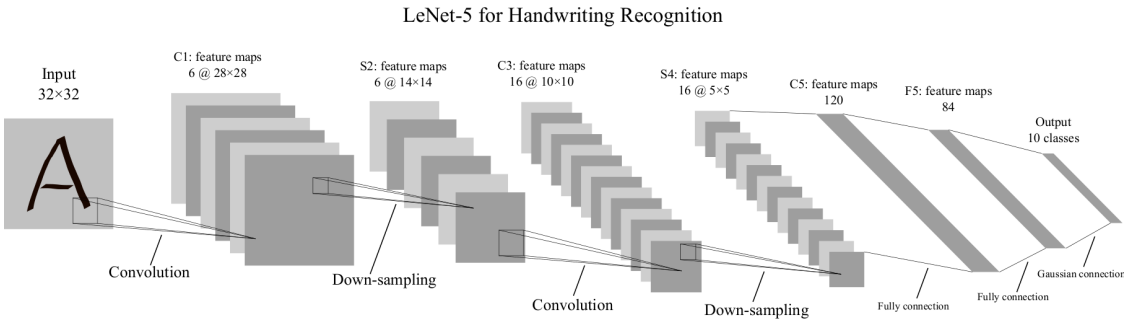


Figure 2.1: A classic CNN architecture, LeNet-5 [61], designed for handwriting recognition.

breakthroughs in certain visual recognition tasks.

Traditional machine-learning models usually design intricate processes that transform training samples into suitable representations or feature vectors, from which a subsequent learning system (e.g. a classifier or a regressor) is then trained to recognize patterns in the input [52]. Deep learning is a class of learning methods that has recently gained popularity. By composing non-linear modules that each transforms the representation at one level (starting with the raw input) into a representation at a higher and more abstract level, deep-learning methods can learn extremely complex functions. The abstraction level goes higher as the representation level goes higher. The key aspect of deep learning is that features from all the layers are not manually designed but are learnt in a fully data-driven fashion [52]. Deep neural networks have found applications in various domains: computer vision [53], natural language processing [54], speech recognition [55], robotics [56], human-level gaming agent [56–58], high-energy physics [59], bioinformatics [60], etc. Deep learning is expected to have more successes in the future because it requires little hand-crafted engineering, so it can easily take advantage of advances in data acquisition and computational efficiency [52]. One particular class of deep-learning models, the convolutional neural network (CNN), has drawn much attention recently. It has achieved great successes in the area of image recognition [53] and has been applied to process data that comes in the form of multiple arrays such as 2D images with red-green-blue (RGB) color channels or 3D medical imaging volumes.

2.1.1 Computational Layer of CNN

A representative CNN model, the LeNet-5 (designed for handwriting recognition tasks [61]), is shown in Figure 2.1. The input image was fed to a stack of convolutional layers, pooling (downsampling) layers and fully connected layers that eventually transformed a structured input (i.e. an image) into a vector of class scores. It is worth noting that modern CNN architectures do not operate in a fundamentally different manner from the LeNet-5, which was proposed in 1998. Nevertheless, numerous hardware-related and software-related efforts have been made to refine CNN's computational framework in order to achieve more efficient and effective machine learning that can handle data of much larger scales than that of LeNet-5's MNIST dataset[1]. This section introduces some of the most representative computational layers that are widely adopted for a variety of image analysis tasks, including those presented in this thesis.

Convolutional Layer

The convolutional layer is the core building block of any CNN. Each layer usually contains a set of convolution kernels (also known as filters) $\{K_i, i = 1, \cdots, N\}$, with which convolution operations are performed on an input multidimensional array of data. A kernel is a multidimensional array of parameters that are adaptable using learning algorithms. The number of kernels N is often referred to as the *channel number* of a convolutional layer. For example, let I denote an input 2D image. The following convolution operation is performed independently for each kernel $K_i \in \mathbb{R}^{m \times n}$ of a convolutional layer to obtain a set of 2D outputs $\{O_i, i = 1, \cdots, N\}$:

$$O_i(x, y) = (K_i * I)(x, y) = \sum_m \sum_n I(x - m, y - n) K_i(m, n) \tag{2.1}$$

The outputs are often referred to as the feature maps. This operation can be implemented in a sliding window approach. The following parameters control the behaviour of convolution:

- **stride**: the amount by which the kernel shifts at a time.

[1] `http://yann.lecun.com/exdb/mnist/`

- **padding**: the amount of zero-paddings on inputs when reaching boundaries.
- **dilation**: the spacing between kernel parameters (also known as the à trous algorithm).
- **group**: the kernel set is divided into groups and convolutions are performed in a group-wise manner.

Deep learning systems have benefited a lot from key properties of convolutional layers such as sparse interactions and parameter sharing [52, 62]. CNNs typically have sparse interactions, which means every output unit only interacts with a portion of the input since kernels are of smaller sizes than the input. Through a sequence of operations that increases the size of receptive fields such as pooling and dilation, CNN can efficiently promote indirect interactions between units in deeper stages and a large portion of the input without an overwhelmingly large set of parameters. All units in the same feature map share the same kernel, hence the idea of parameter sharing. Such a design enables the formation of local distinctive feature patterns and provides a CNN with a position-invariance property [52]. For detailed descriptions of convolution layers, please refer to Goodfellow et al [62].

Pooling Layer

The role of pooling layers is to merge semantically similar features into one. This is usually achieved by only keeping the maximum or mean of a local patch of units in one feature map, which corresponds to a max-pooling layer and an average-pooling layer respectively. Consequently, a convolutional layer and its subsequent pooling layer have the same number of feature maps but usually different spatial resolutions due to the down-sampling effects of pooling layers. The idea of pooling creates invariance to small shifts and distortions in a CNN. Like other deep-learning methods, CNN exploits the property that many natural signals are compositional hierarchies, in which higher-level features are obtained by composing lower-level ones with an increasing degree of abstraction [52]. To build up such hierarchies, a CNN usually contains a stack of convolutional layers and pooling layers to gradually enlarge the

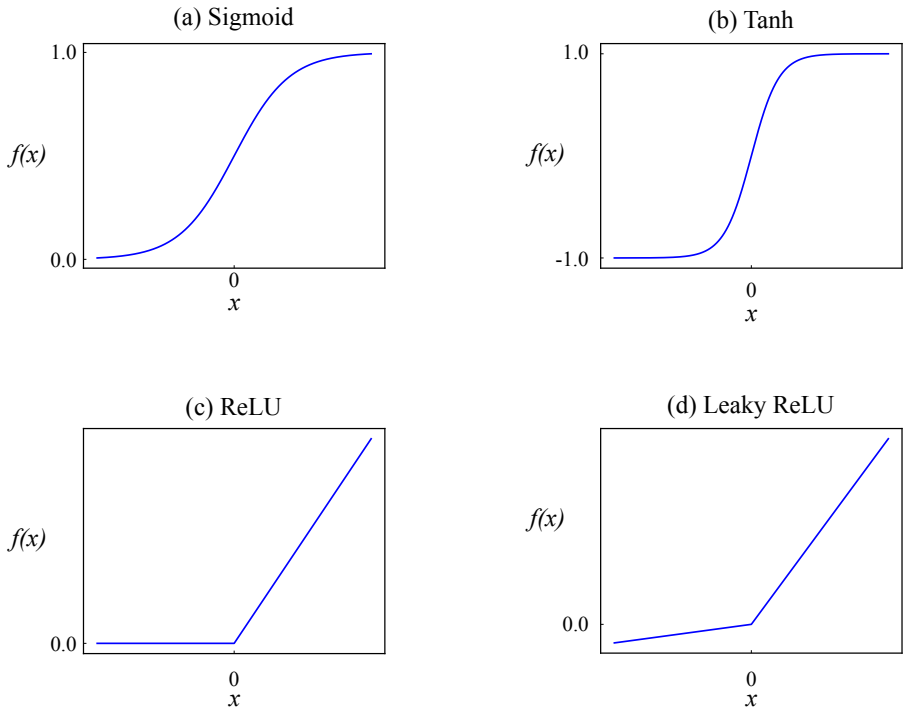


Figure 2.2: Four commonly used activation functions.

network receptive fields in order to acquire higher-level semantics. It is worth noting that while a CNN architecture is inspired by the LGN-V1-V2-V4-IT hierarchy in the visual cortex ventral pathway in the human visual neuroscience research [63, 64], units in a CNN work in a completely different and rather tenuous way when compared to the intricate and complicated way that neurons operate in human brains.

Activation Function

The activation function of a unit in a CNN defines the mapping from an input to the output of the unit. Such a mapping is usually non-linear and continuously differentiable such that the CNN can compute non-trivial problems via gradient-based learning. This thesis applies the following four types of activation functions.

- Sigmoid function. A sigmoid function maps a real number to a value between 0 and 1 exclusively, which is often used to model probabilistic behaviours. As shown in Figure 2.2a, it is given by:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.2}$$

with the following derivative:

$$f^{'}(x) = f(x)(1 - f(x)) \tag{2.3}$$

- Hyperbolic tangent function (Tanh). A hyperbolic tangent activation function maps a real number to a value between -1 and 1 exclusively. It has been reported to perform better than the sigmoid function in terms of gradient saturation [62]. As shown in Figure 2.2b, it is given by:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.4}$$

with the following derivative:

$$f^{'}(x) = 1 - f(x)^{2} \tag{2.5}$$

- Rectified linear activation function (ReLU). This is the default activation function recommended for use with CNNs [62], due to the fact that its unsaturated gradient when $x > 0$ speeds up training in general. As shown in Figure 2.2c, ReLU is given by:

$$f(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases} \tag{2.6}$$

with the following derivative:

$$f^{'}(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ 1 & \text{for } x > 0 \end{cases} \tag{2.7}$$

- Leaky Rectified linear activation function (Leaky ReLU [65]). Leaky ReLU allows a controlled positive gradient when $x < 0$. As shown in Figure 2.2d, Leaky ReLU is given by:

$$f(x) = \begin{cases} 0.01x & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases} \tag{2.8}$$

with the following derivative:

$$f^{'}(x) = \begin{cases} 0.01 & \text{for } x \leq 0 \\ 1 & \text{for } x > 0 \end{cases} \tag{2.9}$$

Normalization Layer

A lot of issues may occur when training a deep CNN model using gradient-based learning algorithms. One notable issue is known as vanishing gradient problem. It can be attributed to accumulation of saturated gradients, which makes the training very slow. Batch normalization [66] is a method of adaptive reparameterization that recalibrates network activations across a mini-batch to have zero-mean and unit variance, followed by an affine transformation with adaptable parameters. This method greatly reduces the difficulty of training a CNN and initiates a family of normalization methods that aim to optimize the learning process, such as instance normalization [67], weight normalization [68], layer normalization [69] and group normalization [70], just to name a few.

2.1.2 Loss Function and Optimization

Recent advances in deep-learning-based medical image analysis studies have been primarily driven by supervised learning algorithms. Roughly speaking, supervised learning algorithms are learning algorithms that map some input to some output, given a training set of inputs $\mathbf{x}$ and their associated outputs $\mathbf{y}$. $\mathbf{y}$ are often referred to as supervision signals of $\mathbf{x}$. In medical image analysis, collecting $\mathbf{y}$ often requires human experts with certain speciality or domain knowledge.

The loss function used by a supervised learning algorithm measures errors that the learning algorithm makes when predicting the associated outputs of inputs $\mathbf{x}$. In practice, the loss function of a mini-batch is often written as the average of the per-sample loss across the mini-batch:

$$J(\theta) = \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p_{data}} L(\mathbf{x}, \mathbf{y}; \theta) = \frac{1}{m} \sum_{i=1}^{m} L(\mathbf{x}_i, \mathbf{y}_i; \theta) \tag{2.10}$$

where L is the per-sample loss, m is the size of the mini-batch and θ denotes the set of trainable parameters. Equipped with a proper loss function, a CNN can be easily trained through back-propagation using stochastic gradient descent algorithms [61,

71]. The expectation of the gradient can be estimated using gradients computed from the mini-batch, which is given by:

$$\tilde{\nabla}_\theta J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \nabla_\theta L(\mathbf{x}_i, \mathbf{y}_i; \theta) \tag{2.11}$$

The stochastic gradient descent algorithm then updates model parameters:

$$\theta \leftarrow \theta - \alpha \tilde{\nabla}_\theta J(\theta) \tag{2.12}$$

where α is a weighting parameter known as the learning rate. In practice, the original stochastic gradient descent algorithm is often replaced by its accelerated variants such as stochastic gradient descent with Nesterov momentum [72], Adagrad [73], Adam [74] and AdamW [75].

2.1.3 Regularization Methods

Much like training most machine-learning algorithms, training CNNs requires certain forms of regularization of model parameters to fight against over-fitting and improve generalizability. This part reviews several regularization methods and tricks that have shown good performance in practice.

Weight Decay

L-p regularization is a classic method to reduce over-fitting by adding an extra term in the loss function that prevents the model weights from getting large. The commonly used L-p regularization are L1 and L2 regularization. L1 regularization is also known as Lasso regression with the following overall loss function:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} L(\mathbf{x}_i, \mathbf{y}_i; \theta) + \lambda \sum_j |\theta_j| \tag{2.13}$$

while L2 regularization has the following overall loss function:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} L(\mathbf{x}_i, \mathbf{y}_i; \theta) + \lambda \sum_j \theta_j^2 \tag{2.14}$$

where λ is a hyper-parameter known as weight decay. Increasing the penalty for large weights can reduce model complexity and reduce over-fitting.

Dropout

Dropout is a regularization method that has been widely adopted in CNNs [76], especially when fully connected layers take up most of the parameters, making the model prone to over-fitting. A dropout layer is often placed next to a fully connected layer. During a training iteration, individual units are dropped out with a predefined probability, i.e. these dropped units do not participate in the current feed-forward computation as well as back-propagation as if they were completely removed. After this iteration, all dropped units are reinserted with their original weights and the drop-out process starts over. Intuitively, only a subset of network parameters are involved at each iteration. This regularizes the model to learn more robust features related to the task that would generalize better to unseen data.

Data Augmentation

Data augmentation is a regularization strategy that aims to increase the diversity of training data without actually collecting new data. A wide range of data augmentation techniques have been introduced in visual recognition studies, including random cropping, padding, a variety of affine transformation, intensity jittering [53] and adversarial sample generation [62, 77, 78].

Early Stopping

Early stopping is an heuristic strategy to reduce over-fitting. It works by setting a patience level κ. Training will be halted if no performance improvement in terms of some evaluation metrics was observed on a validation set for κ consecutive epochs.

2.1.4 Fully Convolutional Network

CNNs were originally designed for classification tasks, such as digit recognition [61] and natural image classification [53]. In practice, an input image is mapped into a softmax output vector, whose elements measure the confidence that this input image should be classified to certain classes. Efforts have been made to modify CNN architectures to achieve end-to-end pixel-level classification, also known as

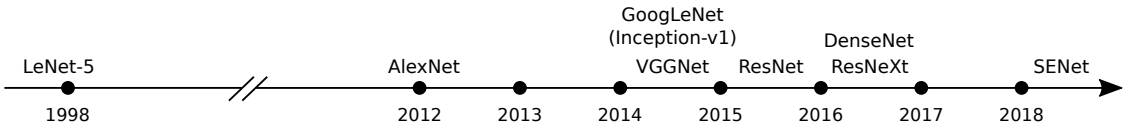


Figure 2.3: Classic CNN architectures in the task of image classification.

semantic segmentation. Recently, a fully convolutional network (FCN) architecture has been proposed for semantic segmentation of natural images [79] and achieves the state-of-the-art in a large-scale segmentation challenge (PASCAL VOC[2]). The key feature of an FCN is that all fully connected layers are replaced by 1×1 convolutional layers that retain spatial information from previous layers. Due to the spatial property, FCNs have been applied to various localization tasks such as one-stage human pose estimation [80], crowd counting [81] and object detection [82].

2.1.5 Classic CNN Architectures for Image Analysis

Over the last decade, numerous CNN architectures have been proposed for different image analysis tasks. One of the most influential tasks is arguably the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Its classification task involves roughly 1.2 million training images of 1,000 semantic categories. From 2012 to 2018, ILSVRC have incubated a variety of CNN-based architectures such as AlexNet [53], VGGNet [83], ResNet [84] and GoogleNet [85]. An illustrative timeline of these architectures is shown in Figure 2.3. This section reviews three classic CNN architectures that are relevant to this thesis.

VGGNet: Proposed in [83], VGGNet showed that convolution kernels with small receptive fields such as 3×3 were well-suited for large-scale image classification tasks. Moreover, VGGNet-based pretrained models have been widely used in transfer learning [83], which facilitates research in representation learning.

ResNet: The deep residual network (ResNet) was proposed in 2016 [84] and have become one of the most cited architectures in the field. A residual learning structure was proposed to solve the performance degradation problem caused by depth increase. ResNet won the ILSVRC 2015 classification task.

[2] `http://host.robots.ox.ac.uk/pascal/VOC/`

SENet: The Squeeze-and-Excitation Networks (SENet) contains a sequence of Squeeze-and-Excitation (SE) blocks that recalibrates channel-wise features in order to boost representational power of a CNN at minimal additional computational cost. SENet won the ILSVRC 2017 classification task.

2.2 Deep-learning-based Ultrasound Image Analysis

This section reviews deep-learning-based ultrasound image analysis studies that covers three types of tasks: classification, detection and segmentation for a variety of anatomical structures such as fetus, heart, liver and prostate. Note that this chapter only identifies a small part of representative techniques and research topics. The most relevant studies to each technical chapter of this thesis will be further discussed at the *background and motivation* sections of those chapters.

2.2.1 Classification

Yu *et al.* proposed a 16-layer CNN model to automatically recognize fetal facial standard plane in ultrasound imaging [86]. Given a limited amount of training data, they exploited transfer learning to accelerate training process and performed data augmentation to alleviate the over-fitting problems. Han *et al.* built up a biopsy-proven benchmarking dataset containing more than seven thousands ultrasound breast images for breast lesion classification [87]. A GoogLeNet was trained on this dataset to discriminate between benign and malignant tumors [85]. Experimental results showed that the model achieved more than 0.9 area under the received operating characteristic curve (AUC), suggesting that it can be used to assist radiologists in diagnosing breast lesions. Wu *et al.* investigated the task of fetal ultrasound image quality assessment by formulating it as a classification problem and addressing it with CNNs [88]. The proposed method involved two CNN models: an L-CNN localized regions of interest in the fetal abdominal region while a C-CNN assessed the image quality based on L-CNN's predictions. Performance of the proposed method was compared to a panel of three clinicians with relevant

speciality. Experimental results suggested that the proposed method could match the performance of clinicians.

2.2.2 Detection

Yap *et al.* developed three deep-learning-based methods for breast lesion detection in ultrasound images [89]: a patch-based LeNet [61], a U-Net [90] and an FCN based on the ImageNet-pretrained AlexNet [53]. Experimental results from two independent datasets demonstrated the superior performance of deep-learning-based approaches over several state-of-the-art lesion detection algorithms. Baumgartner *et al.* achieved real-time detection and localization of 13 fetal standard views in freehand 2D ultrasound images using a CNN model [91]. Different from traditional supervised learning, the network learnt to localize fetal structures and generate detection bounding boxes using only image-level labels. Experiments on real-time annotation, retrospective frame retrieval and localization on a large dataset demonstrated the good performance of their proposed method.

2.2.3 Segmentation

Yang *et al.* addressed the issue of boundary incompleteness encountered in automatic prostate segmentation from ultrasound images [92]. A recurrent neural network was used to learn shape priors from the dynamic sequences of serialized static 2D prostate ultrasound images. A multi-view fusion method was presented to merge shape predictions obtained from multiple perspectives. Shape predictions were further refined by an Auto-Context scheme. Collectively, this framework was shown to resolve boundary incompleteness issue and achieve high performance in prostate segmentation. Anas *et al.* automated the delineation of prostate boundaries in transrectal ultrasound images by using a residual CNN [84] with dilated convolutional layers at deeper stages. The obtained model achieved a Dice similarity coefficient of 0.94 and outperformed two previous state-of-the-art methods [93]. Wang *et al.* developed a 3D CNN model to address 3D prostate segmentation in transrectal ultrasound volumes [94]. Attention modules were

used to merge complementary information encoded in different layers and refine multi-level features in a completely data-driven manner in order to suppress non-prostate noise at shallow layers. Experimental results showed the proposed feature aggregation method achieved accurate prostate segmentation. Looney *et al.* trained the DeepMedic model to segment the first trimester placenta from 3D ultrasound volumes. The reference segmentation masks came from a semi-automated random walker method [95–97]. Experimental results suggested that the resulting model could potentially be used to predict pregnancy that was at risk.

2.3 Summary

Recent years have seen a number of breakthroughs in the field of medical image analysis that are driven by deep-learning-based approaches. Nevertheless, challenges such as lack of training data and model over-fitting are still encountered. While many studies focus on detection and segmentation of certain anatomical structures, very few follow-up work have been reported to develop a complete, image-based diagnosis pipeline based on results from detection and segmentation models. Overall, analysis of placental ultrasound images is less studied. To the best of the author's knowledge, this thesis represents the first work to date to develop image-based assessment pipeline for placental implantation abnormality.

3

Oxford-Hackensack Database of Placenta Accreta Spectrum Disorders

Contents

3.1 Ultrasound Scanning Protocol 31
3.2 PASD-related Annotations 33
3.2.1 PASD Screen Label 34
3.2.2 Image Information Label 35
3.2.3 Placental Lacunae Dot Annotation 35
3.2.4 Utero-placental Interface Labelling 38
3.2.5 Placenta Previa Screen Label 40
3.3 Relation to Each Chapter 42

A placental ultrasound image analysis database used in this thesis is introduced here, namely the Oxford-Hackensack database of PASD. This chapter describes the scanning protocols and illustrates different types of PASD-related annotations, followed by a summary of the database's relation to the subsequent chapters.

3.1 Ultrasound Scanning Protocol

The Oxford-Hackensack database of PASD contains 101 *annotated* 3D B-mode ultrasound volumes from 101 subjects, who were recruited from two sites: 83 subjects from the Fetal Medicine Unit, John Radcliffe Hospital (Oxford, United

Kingdom) and 18 subjects from the Center for Abnormal Placentation at Hackensack University Medical Center (New Jersey, United States). These ultrasound scans were originally collected as part of a research project testing power Doppler markers for diagnosis purpose [40, 50, 51]. All the data from the John Radcliffe Hospital was gathered by Dr. Sally Collins, who is a consultant obstetrician and subspecialist in maternal and fetal medicine.

Subjects were referred by their primary health care providers if they suspected high risks of PASD: diagnosis of a placenta previa and a history of previous uterine surgery or sonographic findings suggesting abnormal placenta invasion on routine pregnancy examination. Written informed consents were obtained with local research ethics approval. Exclusion criteria were multiple gestation, women younger than 16 years old (United Kingdom) or 18 years old (United States) and inability to provide informed consent.

Subjects underwent diagnostic imaging according to the local unit protocol. Static, trans-abdominal 3D B-mode ultrasound volumes as well as 3D power Doppler ultrasound volumes[1] of the placental bed were obtained according to a predefined protocol with the subject in a semi-recumbent position and a full bladder using a RAB4-8-D 3D/4D curved array abdominal transducer (4-8,5 MHz) on a Voluson E8^{TM} (GE Healthcare Ultrasound, Wisconsin, United States). Machine settings followed a pre-determined configuration used in a spiral artery flow study. See Collins *et al.* [51] for the complete settings. The subject was asked to remain as still as possible during the scan. Initially a standard 85° volume was collected in the sagittal plane, midline under the uterovesical fold (presumed site of previous caesarean delivery scar and hence most likely site of placenta invasion). Afterwards, three to five volumes were systematically collected until the entire basal plate of the placenta had been scanned. All data were anonymously stored using Sonoview^{TM} (Nidau, Switzerland). Since the stored data format (.vol) was not supported by the input/output modules of most modern image processing libraries, they were first fed into a proprietary software tool developed using the Insight Segmentation

[1]Although both B-mode ultrasound and power Doppler ultrasound were collected, only B-mode ultrasound scans were incorporated in the database.

and Registration Toolkit (ITK) [98] and the Visualization Toolkit (VTK) [99] to generate 3D B-mode volumes via spatial sampling, which were then stored in an accessible data format (.mha). The sampling rate was 0.33 mm/voxel.

Scans of 101 subjects were considered, among whom 48 were confirmed histopathologically to have PASD and the rest 53 confirmed to have normal placentation. The median gestation for all subjects was 30 1/7 weeks of gestation, with the first quartile being 28 3/7 weeks of gestation and the third quartile being 33 1/7 weeks of gestation. There was no difference in the gestational age at ultrasonography for those complicated by PASD (median 30 1/7 weeks of gestation, first quartile 28 2/7 weeks of gestation and third quartile 33 3/7 weeks of gestation) and those not (29 5/7 weeks of gestation, first quartile 28 3/7 weeks of gestation and third quartile 32 6/7 weeks of gestation) [40]. Among the visually acceptable scans of each subject, the preference was given to the one scanned at the earliest gestational age since the placenta may be too big to view with ultrasonography in later gestation. Only one scan was selected for each subject. In total, 101 3D B-mode ultrasound volumes were incorporated in the Oxford-Hackensack database of PASD. The median spatial resolution for all volumes were $392 \times 571 \times 553$ voxels, or $129.4 \times 188.4 \times 182.5$ mm. All volumes were sliced along the sagittal planes to obtain in total 58,202 images, which were annotated with PASD-related information, as presented in § 3.2.

3.2 PASD-related Annotations

Images/volumes of the Oxford-Hackensack database of PASD were manually annotated by the author under the guidance of Dr. Sally Collins, except for the PASD assessment results, which were already assessed at delivery and later confirmed by histopathology. This section presents five kinds of PASD-related annotations in association with computational modules in the prenatal screening pipeline for PASD, accompanied by statistics and examples.

3.2.1 PASD Screen Label

Each 3D B-mode ultrasound volume in the Oxford-Hackensack database has an associated *screen label*, representing its clinical assessment result with respect to PASD. The label falls in one of the three categories: (i) normal placenta (NP); (ii) abnormally adherent placenta (AAP); (3) abnormally invasive placenta (AIP). This categorization was devised in consultation with an experienced clinician (Dr. Sally Collins) to be clinically useful in assisting with the choice of delivery management strategies. Previously, various clinical grading systems were proposed to access the severity of PASD [40, 100]. At the time when the data was gathered, the presence and severity of PASD were assessed and graded at delivery according to the clinical grading system proposed in [40] by an attending obstetrician who had experience with more than 10 cases of PASD. Subsequent histopathological examinations served as *gold standards* to confirm the clinical scores. Histopathology was undertaken by senior pathologists in each unit who have specialist expertise in placentology. Recently, FIGO released a new classification system to assess PASD based on the previous one [40, 100]. PASD labels in the database can be mapped to the existing grading systems following Table. 3.1. Among 101 B-mode volumes gathered in the database, there are 53 NP, 18 AAP, and 30 AIP cases.

It is worth noting that the 53 NP cases were at high risks of PASD, although they were later diagnosed to be normal placentation. In practice, image-based diagnosis of these 53 scans are more difficult than those at low risks. For this reason, they can be referred to as *hard negative samples*. Investigating prenatal assessment of PASD on these hard samples is a strength of this thesis, which may lead to discovery of robust visual features that appear to be more discriminative than those obtained from *easy samples*. One potential limitation is in regards to performance evaluation since it would be clinically relevant to test on *easy samples* to explore the effectiveness of the proposed approach in terms of clinical screening. Furthermore, based on the fact that hard samples are more challenging to diagnose, the performance of the proposed approach in distinguishing NP cases from PASD

cases would be expected to be even higher when dealing with *easy samples*. These shall be further studied in future work.

3.2.2 Image Information Label

According to the technical challenges discussed in § 1.3.2, 3D B-mode ultrasound volumes may contain a number of image planes that are not particularly useful for assessing PASD. As displayed in Figure 1.5, some image planes are largely truncated while some are contaminated too much by noise. An important pre-processing step in an automated prenatal screening pipeline would be to determine whether or not an image plane within a 3D volume is informative for the task. To this end, each image plane in the database was manually tagged with an *information label*. This binary label describes an image as being either *informative* (1) or *non-informative* (0). Specifically, an image plane is defined to be *informative* if it clearly depicts the following anatomical structures without obvious truncation: subcutaneous fat, the placenta, and the serosal surface of the uterus. Figure 3.1 displays some informative and non-informative planes from two volumes in the database. Among 58,202 image planes in the database, only 11,166 were marked as being informative, which in turn necessitate the use of a pre-processing module that filters out non-informative planes in order to improve screening efficiency. Annotations in the rest of this chapter are performed solely on the informative planes.

3.2.3 Placental Lacunae Dot Annotation

According to FIGO consensus guidelines on PASD (Table 1.2 [22]), presence of numerous placental lacunae (PL) may indicate placental vascular abnormality complicated by PASD. Illustrated in Figure 1.4a, PL appear to randomly distribute inside the placenta without a regular shape or size. Automated PL localization may contribute to qualitative and quantitative analysis of PASD prenatal screening. To this end, locations of PL were manually annotated to train learning-based PL localizers (see Chapter 5). Particularly, this thesis introduces a less expensive form of PL annotation that refrains annotators from performing the time-consuming

Table 3.1: The mapping relationship between PASD assessment label and the existing clinical grading systems

Screen label	#	**Clinical grading [40]**	**FIGO [100]**	**Clinical criteria**
Normal placenta (NP)	53	Grade 1	Grade 0	Complete placental separation, not PASD.
Abnormally adherent placenta (AAP)	18	Grade 2	Grade 1 (creta)	At vaginal delivery: manual removal of placenta required and parts of placenta regarded to be abnormally adherent by a senior, experienced clinician.
				At laparotomy: no placental tissue seen invaded through the serosal surface of the uterus; manual removal of placenta required for remaining tissue and parts of placenta regarded to be abnormally adherent by a senior, experienced clinician.
Abnormally invasive placenta (AIP)	30	Grade 3	Grade 2 (increta)	At laparotomy: abnormal macroscopic findings over the placental bed: bluish/purple colouring, distension (placental bulge); no placental tissue seen invaded through the serosal surface of the uterus; significant amounts of hypervascularity.
		Grade 4	Grade 3a (percreta)	At laparotomy: abnormal macroscopic findings on uterine serosal surface and placental tissue seen to be invading through the surface of the uterus; no invasion into any other organ, including the posterior wall of the bladder.
		Grade 5	Grade 3b (percreta)	At laparotomy: placental villi are seen to be invading into the bladder but no other organs.
		Grade 6	Grade 3c (percreta)	At laparotomy: placental villi are seen to be invading into the broad ligament, vaginal wall, pelvic sidewall or any other pelvic organ.

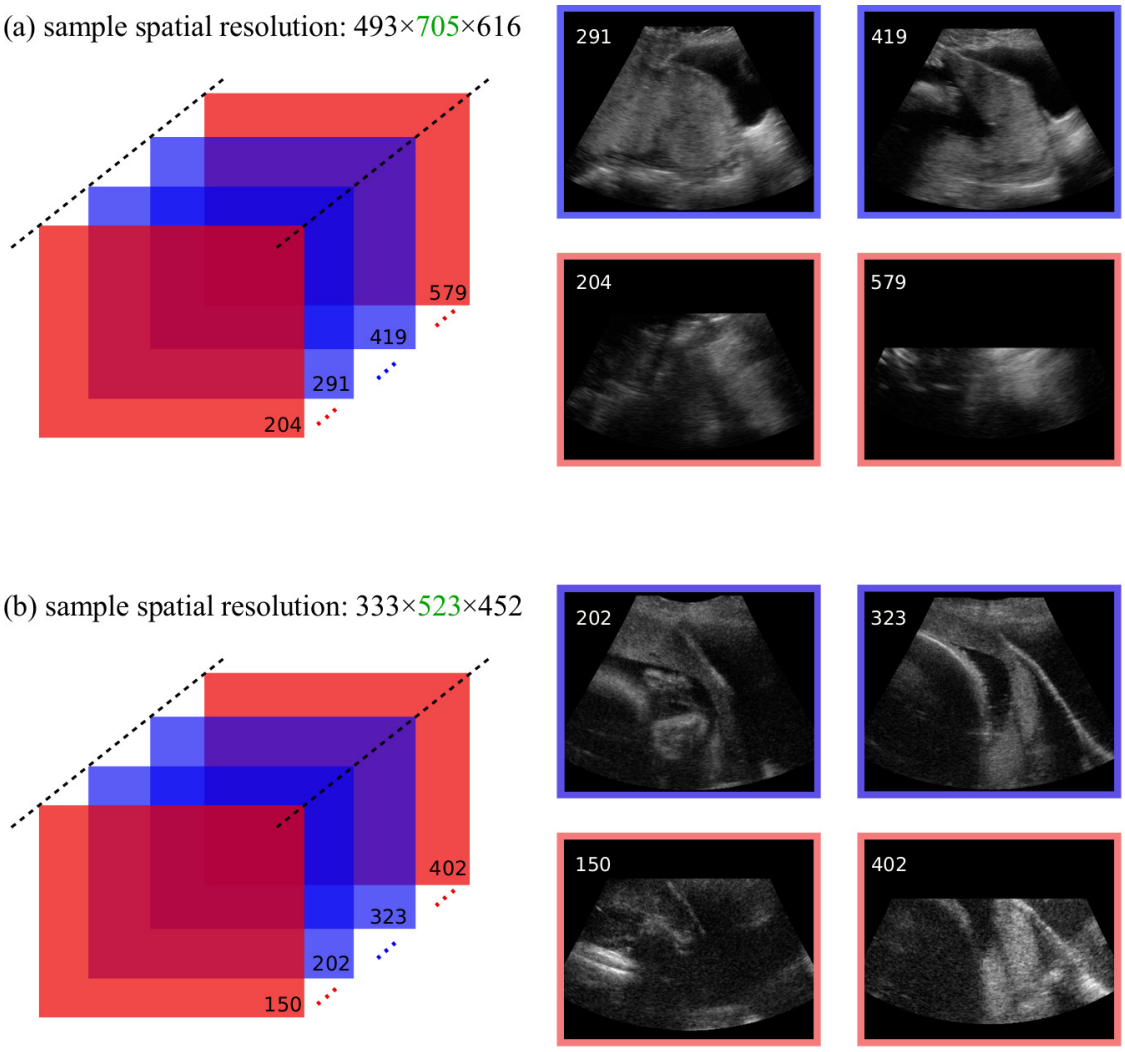


Figure 3.1: Informative planes (blue) and non-informative planes (red) from two volume samples. Plane indices vary along the axis (marked as green) perpendicular to the sagittal plane.

manual segmentation of PL such as delineating PL boundaries. Instead a *dot annotation* is considered. Specifically, the spatial coordinate of the centroid of each identified PL is recorded, which encodes the most reasonable location of PL in the opinion of an expert. Figure 3.2 illustrates samples of dot annotations (red dots) in the database, categorized according to a PASD screen label. Among 11,166 informative planes, 5,008 were identified with PL.

Dot annotations are inclined to subjectivity since the concept of *centroid* may be slightly different across annotators. To overcome this issue and increase interpretability, a new evaluation metric named simplified object keypoint similarity

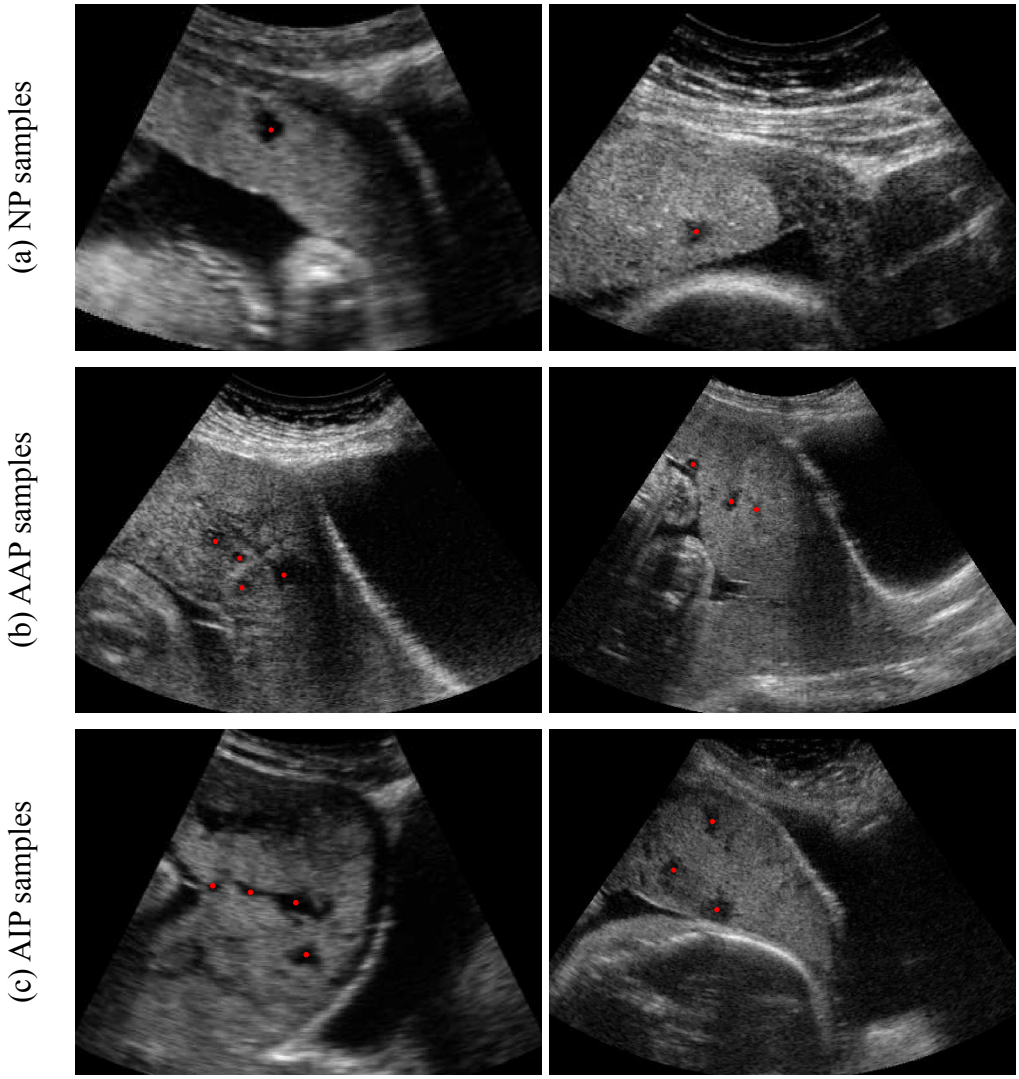


Figure 3.2: Dot annotations for PL instances. Image samples with different PASD screen labels are displayed.

(SOKS) is proposed in Chapter 5 that takes inter-rate reliability into consideration. In general, SOKS is tuned such that a high score is only achieved when a prediction falls into regions where even experienced annotators would likely disagree with each other, rendering the prediction perceptually meaningful. Please refer to § 5.4 for details.

3.2.4 Utero-placental Interface Labelling

Accurate detection of utero-placental interface (UPI) serves as a key step towards automated prenatal screening for PASD due to the fact that sonographic patterns

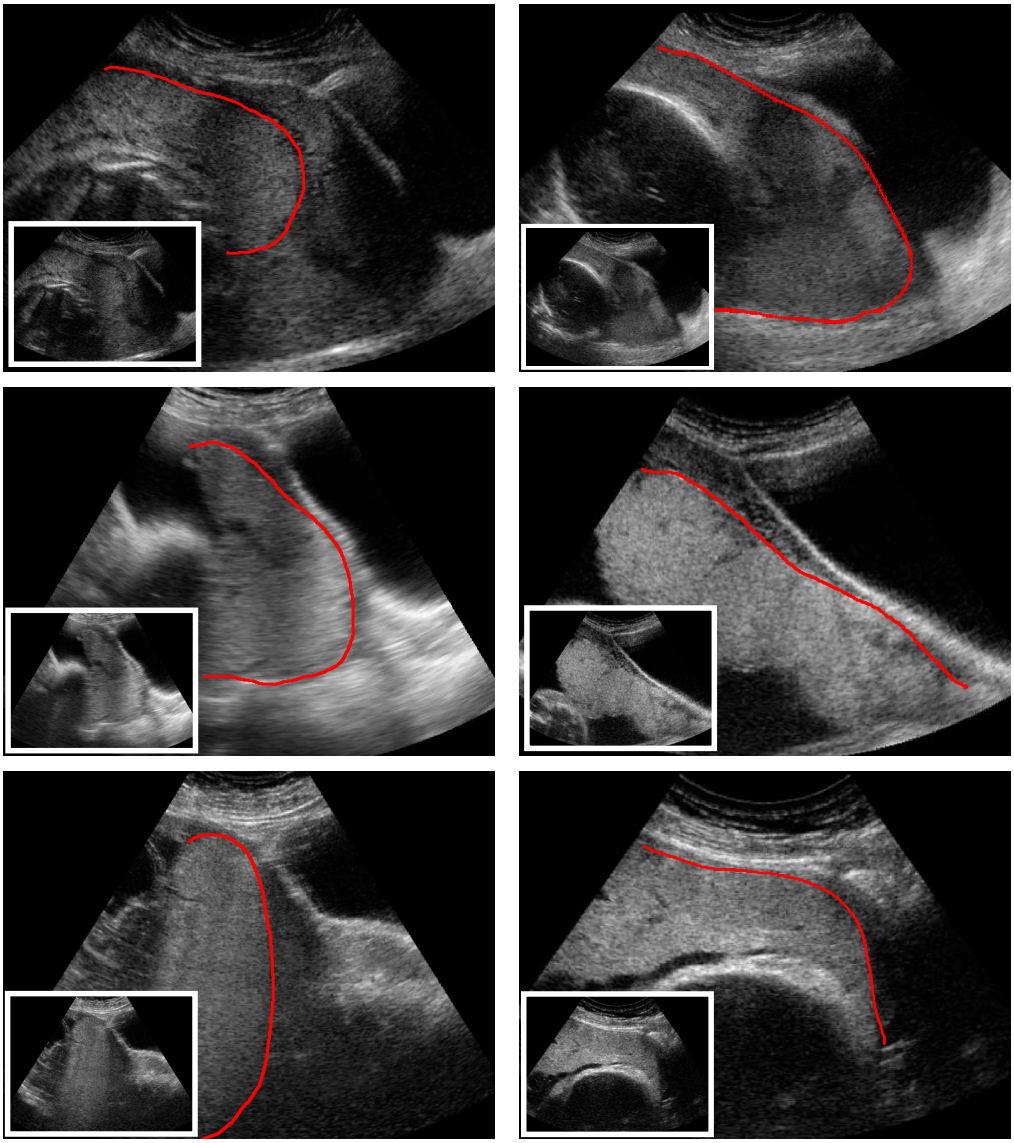


Figure 3.3: Samples with UPI labelling.

on/near UPI may reveal whether the placenta is well separated from the uterus or adheres/invades in an abnormal way. As illustrated in § 1.3.2 (especially Figure 1.4), identifying UPI is a time-consuming task even for experienced sonographers. In particular, local contrast varies along UPI, mainly caused by the indistinct imaging and signal artefacts. As a result, expertise on placenta anatomy is often required to extrapolate UPI in challenging local regions. Moreover, the uterus is constantly growing during pregnancy and shape may be affected by such things as fullness of the bladder during a scan. This results in an inconsistent shape and length of UPI across gestation as well as between subjects. For manual annotation, annotators

Previa screen label	Planes	Volumes	Annotation criteria
No sign of previa	2,140	19	Either internal cervical os is not visible to the annotator or the lower placenta edge is more than 2 cm away from the internal cervical os.
Low-lying placenta	2,292	21	The lower placenta edge is less than 2 cm away from the visible internal cervical os but there is no sign of blocking.
Placenta previa	6,734	61	The placenta visibly blocks the internal cervical os.

Table 3.2: Annotation criteria for previa screen label, with relevant statistics in the database.

relied on global context to first identify the UPI neighbourhood and then delineated the boundary into a curve of 1-pixel (px) width according to local cues, which was stored as a binary mask (i.e. pixels on the UPI take the value 1 and the rest take 0). Figure 3.3 displays several annotation examples. The UPIs of the 11,166 informative planes in the database were manually annotated. The median UPI length is 466 px, ranging from 144 px to 1,113 px. Due to indistinct appearance in placental ultrasound, UPI annotation is inclined to subjectivity. Given a reference UPI mask, a prediction with minor pixel-level deviation from the reference should be not heavily punished in terms of evaluation metrics. Therefore, traditional segmentation metrics such as the Dice co-efficient or the Intersection-over-Union score are not suitable in this scenario. To this end, this thesis adopts standard edge detection metrics [101], namely the best F-measure on the dataset for a fixed prediction threshold (optimal dataset scale, or ODS) and the mean F-measure on the dataset for the best threshold in each image (optimal image scale, or OIS). These metrics provide a tolerance mechanism that avoids punishing small pixel-level deviation. Please refer to § 6.3.3 for details.

3.2.5 Placenta Previa Screen Label

As one of the major risk factors of PASD [19], placenta previa is considered a relevant topic in this thesis. Empirically, a pregnancy case complicated by placenta previa is at risk of PASD according to § 1.2.1. Compared to ultrasonography-based prenatal

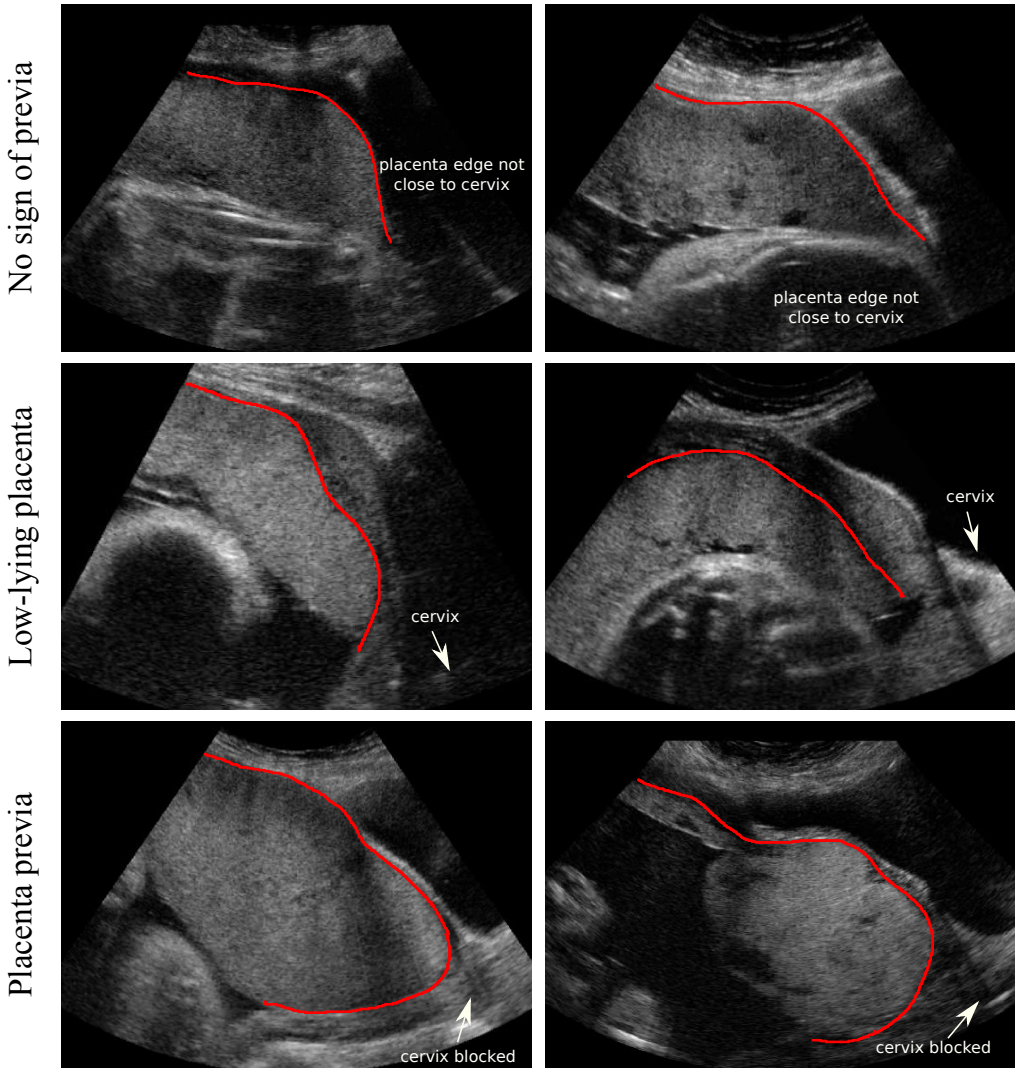


Figure 3.4: Samples with previa screen labels

screening for PASD, placenta previa screening is a relatively less challenging task where the process focuses on examining the relative position of the placenta to the internal cervical os. Following the clinical classification of placenta previa introduced in § 1.1, the author (under the guidance of Dr. Sally Collins) manually tagged a *previa screen label* to each informative plane based on the visual evidence. The annotation protocol is presented in Table 3.2, along with the relevant statistics of the database. Samples with different previa screen labels are displayed in Figure 3.4.

Chapter	Task	Data amount	Evaluation protocol
4	Weakly-supervised localization	10,808 planes from 60 volumes	Subject-level split: training (64%), validation (16%), test (20%)
5	PL localization	3,440 planes from 34 volumes	Subject-level nested 10-fold cross-validation
6	UPI detection	4,871 planes from 49 volumes	Subject-level nested 10-fold cross-validation
7	UPI detection	11,166 planes from 101 volumes	Subject-level nested 6-fold cross-validation
8	Prenatal assessment of PASD	101 volumes	Subject-level nested 6-fold cross-validation

Table 3.3: Data usage in each chapter

3.3 Relation to Each Chapter

Building up the Oxford-Hackensack database of PASD is a progressive process. Data collection was conducted over 6 years (2012 - 2018). As a result, the corresponding data annotation was sequentially performed for 4 years (2015 - 2019). This results in the discrepancy of the amount of data used in the research findings reported in subsequent chapters. The database's relation to each chapter is presented below, which hopefully could eliminate the potential doubt with respect to data usage. Details on data usage are covered in each corresponding chapter. A summary is presented in Table 3.3.

- Chapter 4 presents work on weakly supervised learning of placental ultrasound images, which was mostly conducted in 2016. At that time, image information labels of 60 placental ultrasound volumes were available.

- Chapter 5 presents work on PL localization, which was mostly conducted in 2017. At that time, PL dot annotations of 34 placental ultrasound volumes were available.

- Chapter 6 presents work on UPI detection, which was conducted in 2018, when UPI annotations of 49 placental ultrasound volumes were available.

- Chapter 7 presents work on improving UPI detection via auxiliary learning, while Chapter 8 constructs an image-based prenatal assessment pipeline. Both work were completed in 2019, when PL dot annotations, UPI annotations, and placenta previa screen labels of the entire database (101 volumes) were available.

4

Weakly Supervised Localization in Placental Ultrasound

Contents

4.1 Background and Motivation 45
4.2 Weakly Supervised Localization using ResNets 47
4.2.1 Problem formulation 48
4.2.2 Deep Residual Learning 49
4.2.3 Class Activation Mapping for Localization 51
4.3 Experiments & Discussions 53
4.3.1 Dataset 53
4.3.2 Evaluation Metrics 55
4.3.3 Network Variants 56
4.3.4 Weakly-supervised Localization 59
4.3.5 Discussion 59
4.3.6 Limitations 60
4.4 Conclusions 60

This chapter considers classification algorithms to deal with localizing anatomical structures in placental ultrasound images given only image-level labels. This is referred to as a weakly supervised learning problem[1], and has not been widely studied in medical image analysis.

[1]To prevent ambiguity, the author states that in the context of this thesis, a weakly supervised learning problem refers to localizing anatomical structures using image-level labels (a *weak* form of supervision), as opposed to pixel-level labels (a *strong* form of supervision that is widely used in localization).

A series of deep residual neural networks are developed in order to robustly classify ultrasound patches and extract location-sensitive features.

The main ideas presented in this chapter were first reported at the Annual Conference on Medical Image Understanding and Analysis (MIUA) 2017, Edinburgh, United Kingdom [102].

4.1 Background and Motivation

The vast majority of learning-based localization algorithms in both the natural and medical image analysis literature rely on large-scale *dense* annotations for training models such as deep convolutional neural networks. However, annotating pixel-level masks or instance-level bounding boxes can be expensive and labour-intensive. In particular, manually creating dense reference masks for regions of interest in radiology is often a time-consuming and inefficient process [103]. In contrast, image-level annotations, i.e. presence or absence of objects/regions in an image, are relatively less expensive to obtain. Consequently, this motivates the development of *weakly supervised learning* approaches that learn from cluttered scenes with multiple object instances and perform class-aware localization using only image-level annotations. It is of great research interest to develop weakly-supervised localization approaches that perform comparably to its fully-supervised counterparts since the former can save a considerable amount of time in annotation and can also be less prone to subjectivity.

Anatomical structures in placental ultrasound images can occur in a wide range of appearances, shapes and positions due to differences in placentation, fetal lie, bladder fullness, and probe positioning during scanning (see § 1.2.3). It challenges less-experienced sonographers to robustly interpret the scene irrespective of the variation. To this end, this chapter investigates the challenging problem of weakly supervised learning to classify and localize anatomical structures in placental ultrasound images using only image-level annotations. Specifically, the application of deep convolutional neural networks (CNNs) [52] is explored due

to its effectiveness in modelling structured inputs (e.g. 2D images, 3D volumes) in a data-driven fashion (see Chapter 2).

CNN-based weakly-supervised object localization has recently become a popular research topic in medical image analysis [97]. It has been demonstrated that CNNs originally trained for image classification can also be used to localize objects via analysis of representative features across layers [104–107]. For instance, Simonya *et al.* proposed a visualization technique by computing the gradient of the class score with respect to the input image [107]. The resulting saliency map can pinpoint the location of objects associated with the class label. Oquad *et al.* proposed a method to transfer mid-level image representations and explicitly search for high-score regions [105]. Zhou *et al.* proposed class activation mapping (CAM) to localize regions with discriminative features in an end-to-end fashion [106]. In general, weakly-supervised localization relies only on image-level classification, which is a desirable property that makes object localization a preferable by-product of classification without additional training.

Some success has been achieved using these methods for localizing object instances (e.g. cars, pedestrians) in cluttered scenes (e.g. streets), but it is unclear whether the same problem formulation remains valid in medical imaging where anatomical structures usually constitute a unique geometry. Another potential downside is the smaller scale of training data in medical image applications, since deep learning models are known as being notoriously *data-hungry*.

In the remainder of this chapter, the weakly-supervised anatomical structure localization problem is first re-formulated to be better suited for placental ultrasound. A deep residual network model is then proposed to recognize the predefined multi-structure classes present in a placental ultrasound image, including the placenta, myometrium, subcutaneous tissue. The suitability of the proposed approach is investigated experimentally and discussed at the end of this chapter.

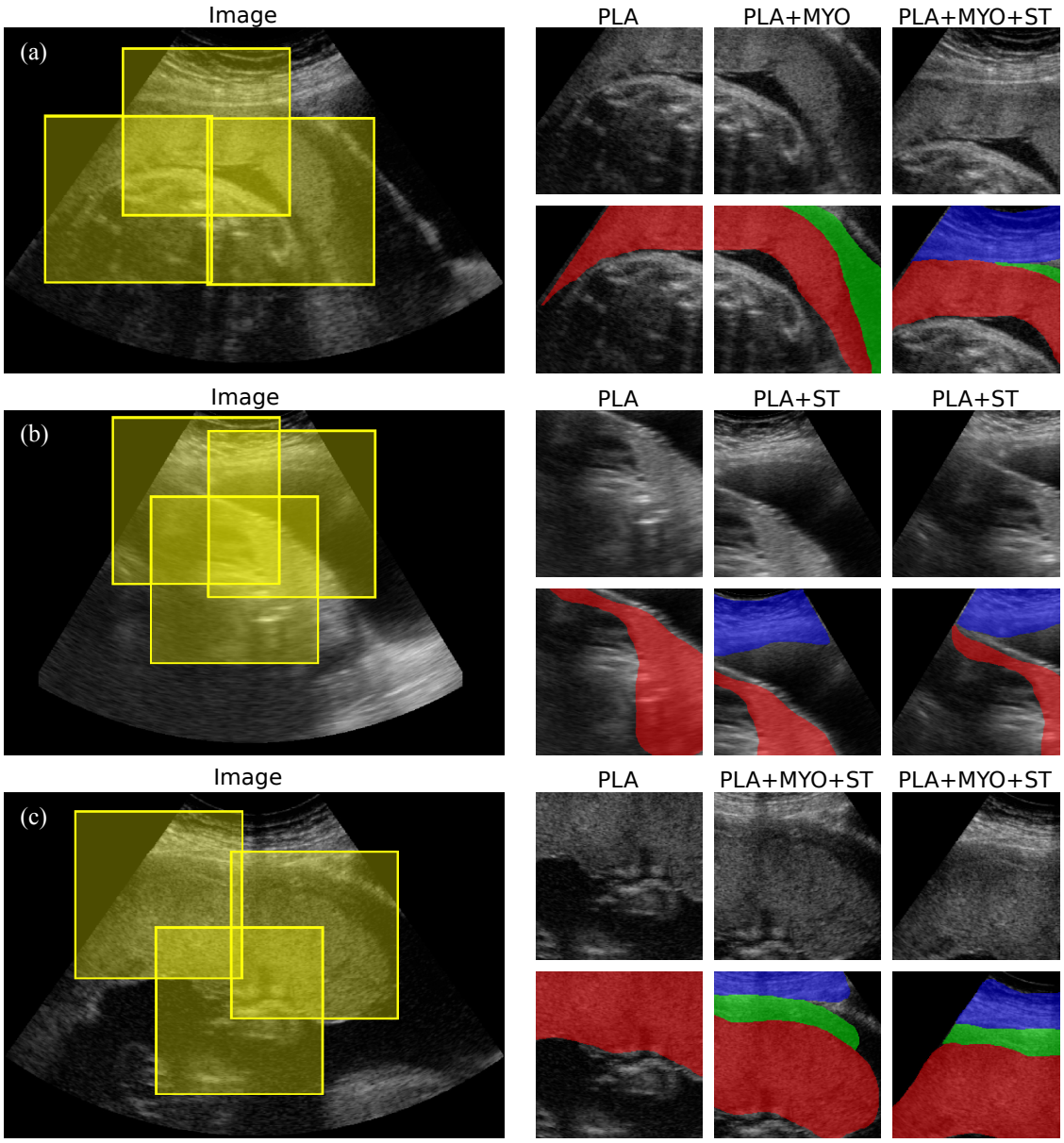


Figure 4.1: Three placental ultrasound image samples taken from the Oxford-Hackensack database. Image patches are randomly cropped and illustrated with reference segmentation masks of the follow anatomical structures: placenta (PLA), subcutaneous tissue (ST) and myometrium (MYO). These patches fall into four multi-structure categories: (i) PLA; (ii) PLA+MYO; (iii) PLA+ST; (iv) PLA+MYO+ST. Please note that all segmentation masks that appear here are used solely for illustration purpose rather than training models.

4.2 Weakly Supervised Localization using ResNets

This section proposes a novel image-level weakly-supervised learning method that promotes localization of discriminative regions across anatomical structures based

on a deep residual network (ResNet) architecture. The aim of the method is to investigate the use of weak supervision in localizing regions of interest in placental ultrasound images without expensive dense annotations.

4.2.1 Problem formulation

In computer vision literature, weakly-supervised learning with image-level labels relies on the co-occurrence of a semantic category across various images to recognize spatially discriminative visual features associated with the category [106]. For instance, distinguishing images of cats from images of dogs will normally result in discriminative features that correspond to cats and dogs respectively. However, this imposed constraint may not be well-suited for placental ultrasound images due to the fact that anatomical structures such as the placenta and the subcutaneous tissues almost always co-occur in an image. It is not realistic to collect images that contain only one structure of interest without the rest cluttering the scene. Therefore, the single-category image-level annotation needs adjusting to better suit the problem.

As displayed in Figure 4.1, randomly cropping a placenta ultrasound image with 224×224 px windows can generate local image patches containing different anatomical structures. The proposed image-level categorization is based on the observation that the vast majority of these image patches fall into one of the four local *multi-structure scenarios*, namely *(i) PLA, (ii) PLA+MYO, (iii) PLA+ST, (iv) PLA+MYO+ST*, where *PLA* denotes the placenta, *MYO* denotes the myometrium and *ST* denotes the subcutaneous tissue. A deep learning model based on the ResNets [84, 108] is proposed to classify placental ultrasound image patches into one of the four multi-structure scenarios. Furthermore, note that the placenta (PLA) is shared by all scenarios, which essentially acts as a distractor for localization. To be discriminative, the model is forced to constitute representative features for each multi-structure scenario, which can then be visualized by CAM, as discussed in § 4.2.3.

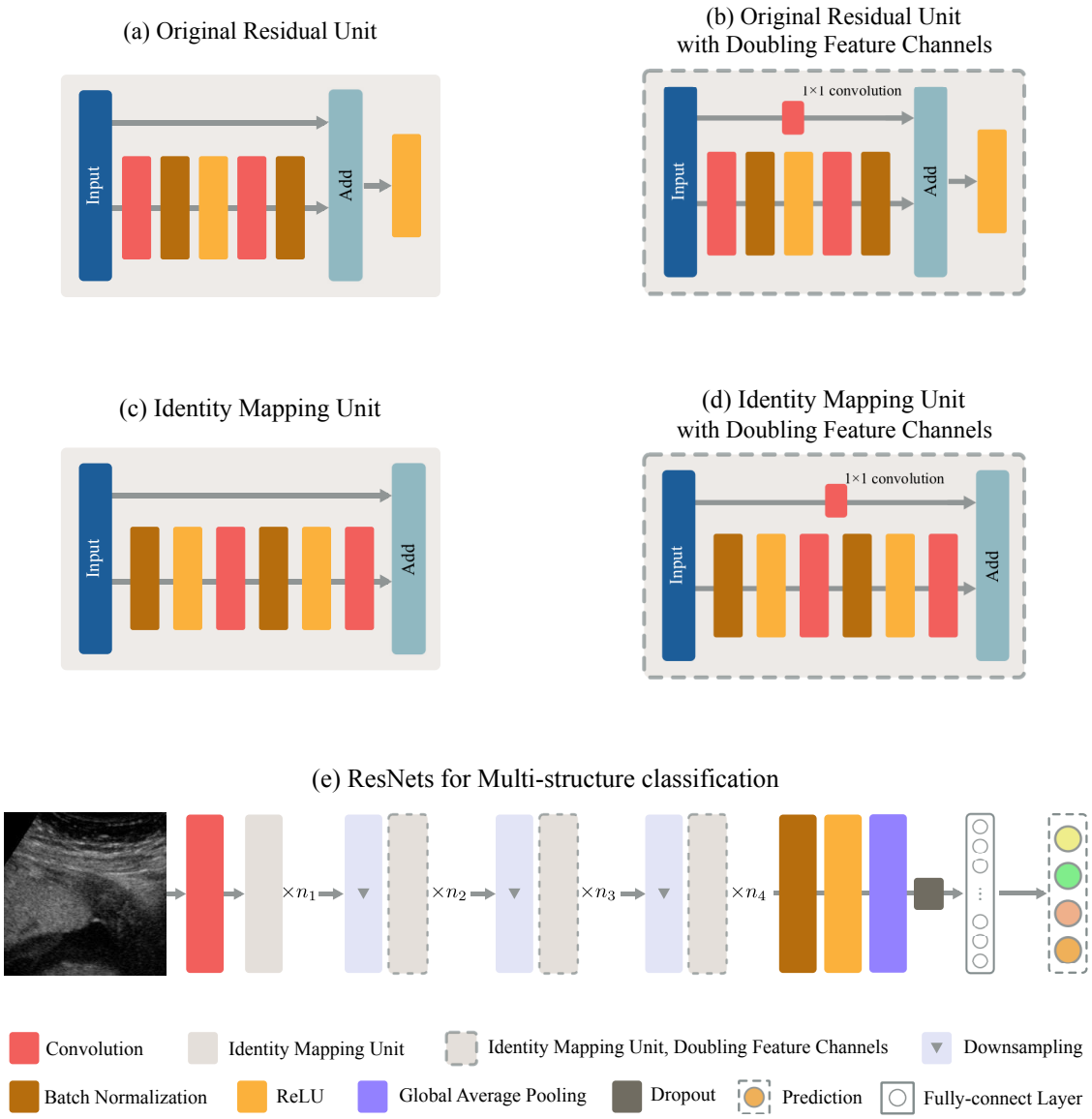


Figure 4.2: Proposed deep residual network architecture with different sizes of residual units, as listed in Table 4.1.

4.2.2 Deep Residual Learning

Deep residual networks (ResNets [84, 108]) have shown impressive representative ability and good convergence behaviours in recent large-scale natural image classification tasks (e.g. ImageNet ILSVRC 2015 [109]), yielding state-of-the-art performance. The original design of ResNets incorporates a sequence of residual units (Figure 4.2a-b). Each unit performs the following computation:

$$\begin{aligned} \mathbf{y}_l &= h(\mathbf{x}_l) + \mathcal{F}(\mathbf{x}_l, \mathcal{W}_l) \\ \mathbf{x}_{l+1} &= f(\mathbf{y}_l) \end{aligned} \tag{4.1}$$

Table 4.1: Network Variants for Placental Multi-structure Classification

Block	Channel	Network Variants (Depth)								
		8	10-A	10-B	10-C	16	18	22	26	26-A
n_1	$16 \rightarrow 16$	1	2	1	1	2	2	2	3	2
n_2	$16 \rightarrow 32$	1	1	2	1	2	2	2	3	2
n_3	$32 \rightarrow 64$	1	1	1	2	3	2	3	3	4
n_4	$64 \rightarrow 128$	0	0	0	0	0	2	3	3	4

where $\mathbf{x}_l$ refers to the input feature to the l-th residual unit and $\mathcal{W}_l$ is a set of weights and biases associated with the l-th residual unit, including weights and biases from convolutional layers and batch normalization layers [66]. Here $\mathcal{F}$ denotes the residual function which is learnt with respect to the input feature $\mathbf{x}_l$. The residual function in Figure 4.2a-b contains a stack of two 3×3 convolutional layers. The function h is either an identity mapping ($h(\mathbf{x}_l) = \mathbf{x}_l$, Figure 4.2a) or a 1×1 convolution (Figure 4.2b) that deals with the channel mismatch when doubling feature channels. The function f is the operation after the element-wise addition. In the original design [84], f is the rectified linear unit (ReLU).

Later, He *et al.* proposed a simple and effective identity mapping unit that enables smooth information propagation through the entire network [108]. Large-scale data experiments revealed that the full pre-activation residual unit, as shown in (Figure 4.2c-d), outperformed the original design by putting batch normalization and ReLU before convolution. This design choice accelerated learning and improved global regularization. An identity mapping unit sets both h and f in Equation 4.1 as identity mapping, i.e. it performs the following computation:

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \mathcal{F}(\mathbf{x}_l, \mathcal{W}_l) \tag{4.2}$$

As a result, such a design allows a recursive derivation:

$$\mathbf{x}_L = \mathbf{x}_l + \sum_{i=l}^{L-1} \mathcal{F}(\mathbf{x}_i, \mathcal{W}_i) \tag{4.3}$$

for any deeper unit L and any shallow unit l, which implies the smooth information propagation throughout the network. Given a predefined loss function $\mathcal{L}$, the

backward gradient computation for Equation 4.3 is also intuitive according to the chain of back-propagation [110]:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{x}_l} &= \frac{\partial \mathcal{L}}{\partial \mathbf{x}_L} \cdot \frac{\partial \mathbf{x}_L}{\partial \mathbf{x}_l} \\ &= \frac{\partial \mathcal{L}}{\partial \mathbf{x}_L} \left(1 + \frac{\partial}{\partial \mathbf{x}_l} \sum_{i=l}^{L-1} \mathcal{F}(\mathbf{x}_i, \mathcal{W}_l)\right)\end{aligned} \tag{4.4}$$

Taking identity mapping units as the basic building blocks, a ResNet-based architecture is proposed for multi-structure classification (Figure 4.2e). In general, this architecture contains four groups of residual blocks, each of which further consists of n_i stacked identity mapping units ($i = 1, 2, 3, 4$). Here $\{n_i\}$ is a set of architecture hyper-parameters that control the entire depth D of the network, where $D = 2(n_1 + n_2 + n_3) + 2$ denotes the number of convolutional layers. Varying $\{n_i\}$ leads to a series of network variants with different representative capacities, as listed in Table 4.1. Following the design choice adopted in recent literature [83, 84, 111], all convolutional layers in the identity mapping units employ a small convolutional kernel of size 3×3, except for the 1×1 convolutional layer in Figure 4.2d. At the beginning of the second, third and fourth residual modules, convolutional layers with stride 2 are deployed to downsample the feature maps. Meanwhile, the convolution doubles the feature channels, yielding an increasing channel sequence: $16 \rightarrow 32 \rightarrow 64 \rightarrow 128$. The last residual module is followed by a global average pooling (GAP) layer and finally a fully-connected layer to output a class prediction vector of length 4. A dropout layer [76] with the dropout probability of $p = 0.3$ is inserted between the GAP layer and the fully-connected layer, which demonstrates a good regularization performance across network variants according to experiments. The use of GAP is described in the following subsection to boost discriminative localization. The proposed ResNets are trained in an end-to-end fashion by the stochastic gradient descent algorithm with a momentum of 0.9.

4.2.3 Class Activation Mapping for Localization

Each class has its corresponding CAM which visualizes discriminative image regions used by the network to identify this specific class. As displayed in the bottom of

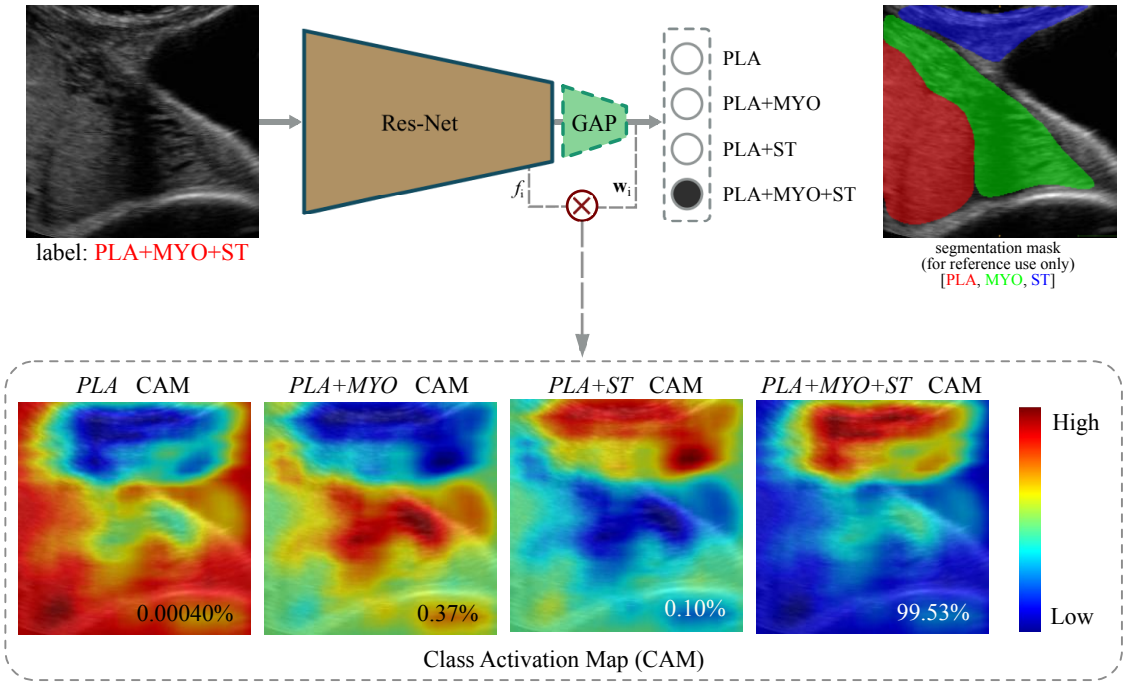


Figure 4.3: Proposed pipeline for multi-structure classification. Given an unseen image patch from a placental ultrasound, the network predicts the multi-structure class within. By computing the class activation maps of each class, the network can localize discriminative regions associated with the corresponding anatomical structures.

Figure 4.3, the CAM for class i is generated by computing a weighted sum of all feature maps from the rectified activation of the last convolutional layer. Here the weights refer to the corresponding weight vector $\mathbf{w}_i$ learnt in the fully-connected layer. Specifically, let M_c denote the CAM for class c. Each spatial element of M_c is given by

$$M_c(x, y) = \sum_i \mathbf{w}_i f_i(x, y) \tag{4.5}$$

where f_i represents the activation of channel i in the last convolutional layer at spatial location (x, y). A simple up-sampling would suffice to map the CAM heat-map (28×28) back to the input size (224×224).

The use of GAP encourages the network to identify the *extent* of the object, rather than pinpointing the object on a *specific location* as global max pooling (GMP) does [105]. It is intuitive that this global averaging operation should boost identification of a local discriminative region in order to reach a lower global loss, while a global maximum operation only influences the maximal value of a feature map [106].

4.3 Experiments & Discussions

This section describes experiments to evaluate predictions of the multi-structure classification models. The proposed ResNet variants are trained on a custom dataset containing image patches cropped from the informative image planes in the Oxford-Hackensack database of PASD (see Chapter 3). Performance evaluation is conducted on a separate test set. Results of the weakly-supervised localization are also displayed. The models are implemented in Torch 7 [112] on a 64-bit Ubuntu 15.04 machine with a NVIDIA graphics card.

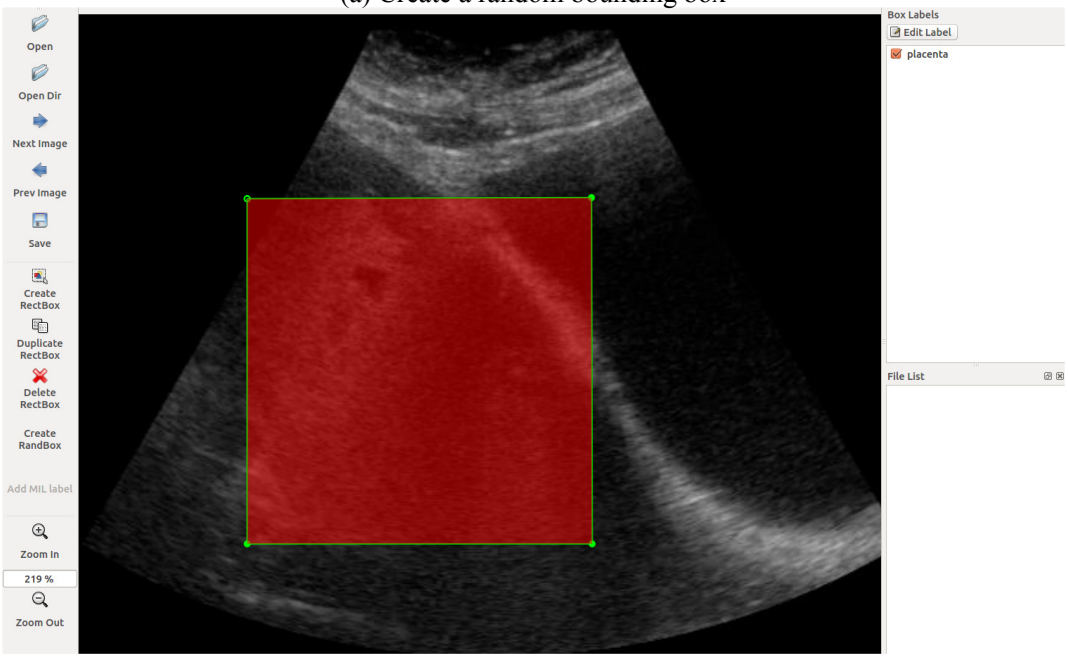
First, the dataset used in all experiments are introduced in § 4.3.1. Then the evaluation metrics used to assess the predictions of the proposed ResNets are introduced in § 4.3.2. Performance analysis of network variants is presented in § 4.3.3. Weakly-supervised anatomical structure localization is further discussed in § 4.3.4, followed by discussions on the suitability of the proposed approach in clinical applications.

4.3.1 Dataset

Informative image planes from 60 placental ultrasound volumes were selected for this task (see § 3.3). In particular, 224×224 image patches were randomly sampled from these image planes and manually tagged with multi-structure labels described in § 4.2.1. The annotation process was performed using a custom software tool *labelMIL* written by the author, which was also based on the widely used *labelImg* annotation tool[2] and has been made freely available[3]. This tool displays each image plane to the annotator and enables arbitrary bounding box generation. The annotator can then label the bounding box by selecting the visible anatomical structures within the box. All the operations can be manipulated with the keyboard shortcut to facilitate efficiency. Annotations were saved as XML files in PASCAL VOC format[4]. There were totally 10,808 annotated image patches, which were then

[2] `https://github.com/tzutalin/labelImg`
[3] `https://github.com/ericqh/labelMIL`
[4] `http://image-net.org/download-bboxes`

(a) Create a random bounding box



(b) Label visible anatomical structures

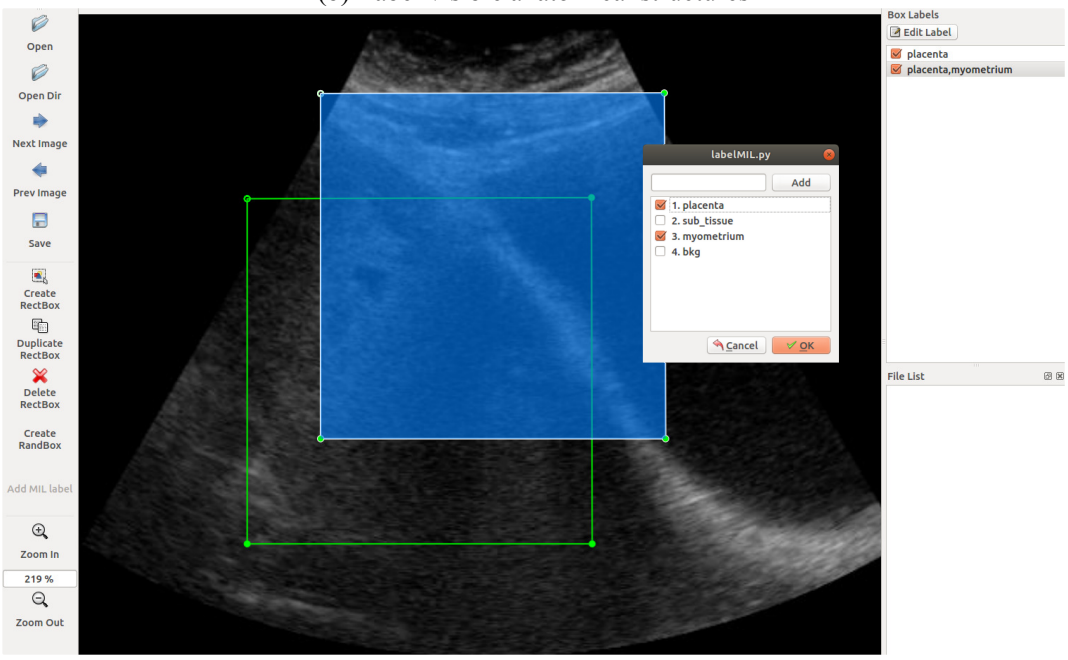


Figure 4.4: A custom annotation tool *labelMIL*.

randomly divided into a training set (64%), a validation set (16%) and a test set (20%) following a patient-level split, as listed in Table 4.2.

Table 4.2: Annotated Image Patches for Weakly-supervised Localization

Category	Training	Val.	Test	Total
PLA	2,764	711	835	4,310
PLA+MYO	1,064	283	342	1,689
PLA+ST	1,834	422	590	2,846
PLA+MYO+ST	1,256	312	395	1,963
Total	6,918	1,728	2,162	10,808

Network Variants	Params. (K)	FLOPs (G)	Mean Top-1 Error Rate (%)	
			Validation Set	Test Set
8	307.22	0.63	18.23	17.81
10-A	311.89	0.87	14.53	14.29
10-B	325.78	0.86	12.85	13.55
10-C	381.20	0.86	10.82	11.29
16	478.42	1.57	11.40	10.78
18	700.24	1.74	9.95	9.20
22	1069.65	2.21	8.56	8.74
26	1092.88	2.68	9.84	8.88
26-A	1439.06	2.67	**8.08**	**8.60**

Table 4.3: Mean top-1 error rate (%) of the proposed multi-structure classification models.

4.3.2 Evaluation Metrics

Standard top-1 error rate is used to evaluate the classification performance of the proposed models. Mean top-1 error rates of ten ResNet variants are presented in Table 4.3 and class-wise top-1 error rates are presented in Table 4.4. It is worth noting that *PLA+MYO* class has the highest top-1 error rate compared to the rest three classes. In placental ultrasound imaging, it is generally difficult to identify the myometrium under various scenarios. In one scenario, the myometrium may not exist at all due to certain placental abnormality (such as PASD). In another scenario, it appears but the texture or intensity is not sufficiently discriminative to be recognized. Moreover, ultrasound signal dropout may hinder a clear visualization. Classification confusion matrices on the test set for the top three models in terms of mean validation error (Model 22, Model 26 and Model 26-A) are presented in Figure 4.5. These models all suffer from the same problem of identifying *PLA+MYO*.

Network Variants	Class-wise Top-1 Error Rate (%) on Test Set			
	PLA	PLA+MYO	PLA+ST	PLA+MYO+ST
8	13.05	26.61	20.85	15.70
10-A	9.82	29.53	12.71	12.91
10-B	8.74	30.41	11.36	12.41
10-C	7.07	23.98	9.15	12.41
16	8.02	22.22	9.15	9.11
18	8.62	19.59	**6.10**	**6.08**
22	8.26	**14.91**	6.44	7.85
26	**6.71**	15.21	8.64	8.35
26-A	7.43	15.21	7.46	7.09

Table 4.4: Class-wise top-1 error rate (%) of the proposed multi-structure classification models.

4.3.3 Network Variants

Architecture hyper-parameters n_1, n_2, n_3, n_4 corresponding to the number of stacked residual units in each residual module, are displayed in Figure 4.2 and Table 4.1. Altering these hyper-parameters enables investigation on how the depth of a ResNet casts impact on the model's generalizability at different abstraction levels. According to the experimental results listed in Table 4.3 and Table 4.4, several arguments can be made in terms of improving classification performance: (i) more residual units should be put in deeper modules that have larger channel sizes and receptive fields, as demonstrated by the better performance of *10-C* and *26-A* compared to their counterparts of the same depth; (ii) the identity mapping units can promote the propagation of information, yielding better performance for deeper networks without causing *degradation* problems, where deeper networks generalize worse to unseen data due to vanishing/exploding gradients [113, 114]; (iii) The class-wise best performance was achieved by model *18, 22* and *26*, as marked in bold font in Table 4.4. The best model in terms of mean top-1 accuracy is model *26-A*. If this is a class priority, a model ensemble approach can be applied to obtain a larger but more accurate model set. Otherwise, model *26-A* is the best single model in the experiment.

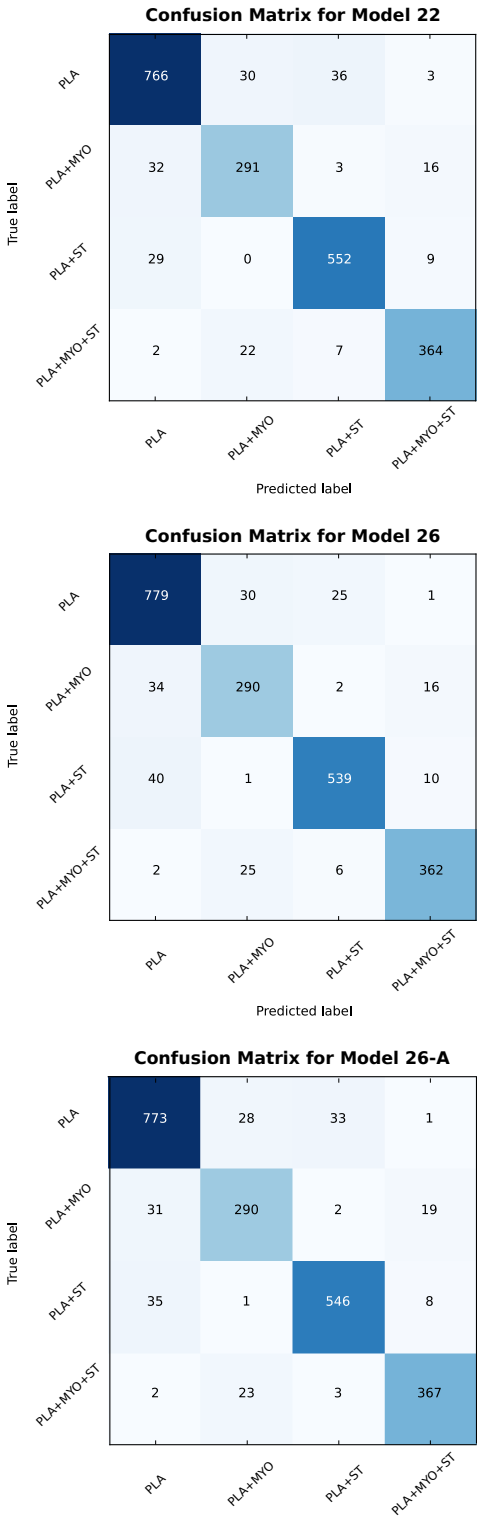


Figure 4.5: Confusion matrices on the test set for the top three models.

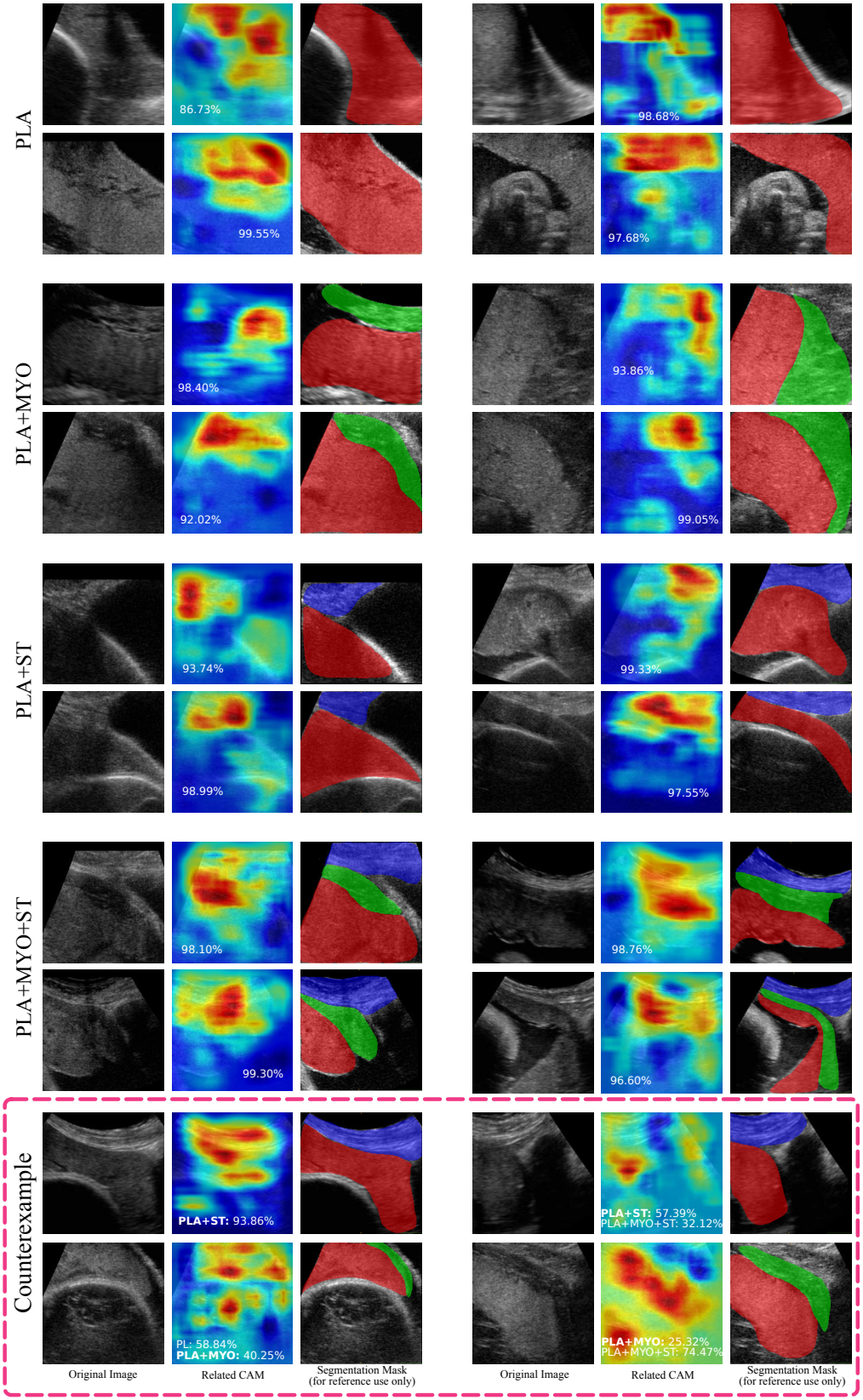


Figure 4.6: Results of weakly supervised localization with reference segmentation mask of the following anatomical structures: placenta (PLA), subcutaneous tissue (ST) and myometrium (MYO). All images are from the test set.

4.3.4 Weakly-supervised Localization

In this subsection, weakly supervised localization of placental anatomical structures based on the top classification model (*26-A*) is analysed. An image patch is first classified into one of the four multi-structure categories, after which the corresponding CAM for the predicted category is generated using Equation 4.5 to highlight the discriminative regions of this category in the image. Some results are shown in Figure 4.6 for each category, where prediction scores as well as segmentation masks are also illustrated. For example in the first triple set, the input image is correctly classified as *PLA* with a score of 0.8673. Its CAM for *PLA* highlights the approximate position of the placenta, as verified by the reference segmentation mask. Some counterexamples are also displayed in the bottom of Figure 4.6, which are either mis-classified (the correct class is labelled in bold font), mis-localized or both. It is observed that good weakly-supervised localization tends to be accompanied by good classification.

4.3.5 Discussion

There are many ways to potentially formulate this multi-structure classification problem. The most intuitive way is probably to build up a *multi-label* learning model by training a group of one-versus-all binary classifiers to identify these anatomical structures respectively. However, this type of model suffers from over-fitting and generalization problems in the data experiments. One possible explanation is that there are strong correlations among these anatomical structures (e.g. myometrium is always co-localized with the placenta), thus it may not be appropriate to model them separately. Moreover, the multi-label learning does not contribute to the generation of a CAM due to the removal of the soft-max normalization. In this chapter, the problem is formulated with the intention to both achieve high classification accuracy and boost weakly supervised localization. Consequently, a multi-structure classification problem is formulated. Intuition behind this formulation is quite straightforward. To be sufficiently discriminative, it is expected that the *PLA+MYO* CAM would highlight the myometrium region

(MYO) since the placenta is likely to appear in the majority of the image patches sampled from a given placenta ultrasound image. For the same reason, the *PLA+ST* CAM is expected to illustrate the region of subcutaneous tissue (ST) and *PLA+MYO+ST* would ideally highlight the joint region of myometrium and subcutaneous tissue. Experimental results confirm the validity of this formulation. CAM demonstrates reasonable localization ability for the corresponding anatomical structures, as shown in Figure 4.6. Furthermore, the classification network appears to use texture as well as boundary information to construct discriminative features. For example, the *PLA* CAM heatmaps typically highlight a portion of the placenta as well as the placenta-background boundaries. Similarly, the *PLA+MYO* CAM heatmaps highlight a portion of the myometrium and the myometrium-placenta boundaries. The *PLA+ST* CAM heatmaps cover the subcutaneous tissue as well as the tissue-placenta interface. Finally, the *PLA+MYO+ST* CAM heatmaps cover regions of both myometrium and subcutaneous tissue, as well as their interface. This is observed across the entire test set.

4.3.6 Limitations

One limitation of this chapter is that quantitative studies of localization performance are missing since relevant manual annotations are not available. All discussions remain qualitative based on visual examination. Pixel-level dense labelling is needed to resolve this issue, which is unfortunately beyond the scope of this thesis. However, the idea of formulating multi-structure anatomy is more relevant in medical imaging compared to image-level classification approaches in computer vision. It is expected that this chapter would provide insight for weakly supervised localization tasks in similar clinical scenarios.

4.4 Conclusions

This chapter has presented a series of deep convolutional neural networks powered by residual units that perform multi-structure classification and weakly-supervised localization in placental ultrasound images. It was found that reasonable localization

can be achieved using carefully formulated image-level annotation without including any spatial knowledge. In particular, it was found that multi-label learning is not suitable for recognizing anatomical structures in placental ultrasound due to correlations among classes. Instead, the clinical observation that certain anatomical structures will always co-occur in placenta imaging provides insight for a multi-structure classification formulation. It was shown that identity mapping units are effective in training deep residual networks to perform accurate multi-structure classification (around 8.6% top-1 error rate). It was found that GAP layers facilitate the spatial expansion of discriminative regions highlighted by a CAM. Furthermore, it was shown that using CAM is highly beneficial for localizing representative textures / boundaries of the corresponding anatomical structures.

As stated in § 1.3.2, accurate classification and localization of anatomical structures is a precursor for automated image-based diagnosis of placental abnormalities. Moreover, it may assist clinicians to better interpret ultrasound images containing indistinct boundaries with low contrast. This chapter has introduced a less expensive approach to develop models that can localize anatomical structures without performing pixel-level dense annotation. An interesting extension to this work would be to jointly analyse *PL+MY+ST* CAM and *PL+MY* CAM, which may provide insight to help further improve localization of the myometrium since they both contain a portion of the anatomical structure.

Recent work (e.g. Gradient-weighted Class Activation Mapping [115]) utilized gradients of target classes to produce coarse localization maps. Furthermore, an ensemble of classification models that achieve best class-wise performance is expected to produce more plausible weakly supervised localization. These shall be investigated in the future.

5

Placental Lacunae Localization via Multi-scale Layer Aggregation

Contents

5.1 Background and Motivation 63
5.2 From Dot Annotation to Confidence Map 67
5.3 Deep Layer Aggregation for Representation Learning 70
5.4 Simplified Object Keypoint Similarity 73
5.5 Experiments & Discussions 77
5.5.1 Dataset 78
5.5.2 Implementation Details 79
5.5.3 Performance Analysis 79
5.6 Error Analysis 83
5.6.1 Error Type 84
5.6.2 Sensitivity to Instance Number 90
5.6.3 Limitation 93
5.7 Conclusions 94

This chapter considers machine learning algorithms that deal with localizing placental lacunae (PL) in placental ultrasound images given a type of *weak supervision*, namely dot annotations (see § 3.2.3). Fully convolutional networks (FCNs) are developed to learn the mappings from input images to PL locations in a bottom-up fashion. Multi-scale features play a significant role in localizing instances with variable shapes and sizes such as PL. Consequently, a number of FCN-based *multi-*

scale feature aggregation approaches are investigated and compared in order to robustly localizing PL given a limited amount of training data.

The main ideas presented in this chapter were first reported at the 21^{st} International Conference on Medical Image Computing & Computer Assisted Intervention (MICCAI 2018), Granada, Spain [116]. This chapter extends the work in that paper by performing a detailed error analysis on the localization results of several competitive benchmarks and the proposed approaches. The analysis is designed to understand the behaviours of these PL localization algorithms and measure their strengths and weaknesses for this challenging task.

5.1 Background and Motivation

Estimating the location of a *keypoint* from an image is challenging in both computer vision and medical image analysis. The most relevant task in computer vision is arguably human pose estimation [80, 117–122], which requires localization of keypoints on the human body (e.g. shoulders, knees, hips) in challenging cluttered scenes. The COCO Keypoint Detection Task[1] was launched in 2016 and has become one of the most influential competitions in human pose estimation. It greatly promotes this field by releasing open-sourced datasets with unified evaluation metrics and guidelines. The latest COCO 2019 Keypoint Detection Task has released more than 200,000 images and 250,000 person instances labelled with keypoints. Nevertheless, localizing keypoints of a person from a single image remains a challenging task due to a number of confounding factors such as variation in perspective projection, occlusion and person-environment interaction. Thanks to recent developments in computer vision research using deep convolutional neural networks, the performance of human pose estimation methods has improved dramatically [123]. As an early effort, Toshev et al. proposed a cascade of CNNs to directly regress the coordinates of body joints given a single-person image [122]. Later, Pischulin et al. explored multi-person pose estimation by first detecting body part candidates and then building up a graph to group them into a complete

[1] `http://cocodataset.org/#keypoints-2019`

body configuration. Cao et al. proposed a more efficient bottom-up approach that learned confidence maps and part affinity fields of body parties simultaneously before matching them in a bipartite graph [80]. Chen et al. presented a top-down detection method to first predict human bounding boxes and then localize body joints using cascaded pyramid networks [120]. A number of recent human pose estimation algorithms, trained on hundreds of thousands of labelled images, have reached more than 100 layers in model capacity. By contrast, many medical image analysis problems do not meet this order of magnitude in terms of data available or model capacity. Therefore, it is of great research interest to investigate keypoint localization approaches that utilize relevant prior knowledge and advanced architectures to improve performance given a limited amount of training data.

To date, keypoint localization studies in medical image analysis literature has covered a number of anatomical structures in different imaging modalities. This section reviews recent work on two types of problems that involve identifying keypoints in medical image data using learning-based methods: (i) anatomical landmark detection; and (ii) instance detection and counting.

(i) Anatomical landmark detection. Anatomical landmarks refer to biologically-meaningful points in an organism. Many clinical applications require the precision detection of anatomical landmarks for subsequent tasks such as organ segmentation [124, 125], image registration [126, 127] and structure tracking [128]. Recent advances in deep learning has seen considerable improvements in learning-based anatomical landmark detection. Zheng et al. proposed an efficient deep learning cascade to detect landmarks in 3D volumetric data and tested their model on carotid artery bifurcation detection on a head-neck CT dataset from 455 patients. A shallow network with only one hidden layer was used for initial voxel screening, followed by a deep classification [129]. Payer et al. exploited heatmap regression using CNNs to enable multiple landmark localization in medical image data. They evaluated the proposed method on a 2D hand X-ray dataset with 37 characteristic landmarks and a T1-weighted 3D gradient echo hand magnetic resonance dataset with 28 annotated landmarks

[130]. Zheng et al. proposed a two-stage landmark detection method to deal with the situation where only limited training data is available [131]. The first stage learned inherent associations between local image patches and target landmark, while the second stage predicted landmarks' coordinates. Their applications included detection of brain landmarks using a 3D T1-weighted magnetic resonance dataset and detection of prostate landmarks using a 3D computed tomography dataset. Ghesu et al. presented a method that reformulated landmark detection problem as a behavior learning task [132]. The proposed deep reinforcement learning method achieved a high detection accuracy in a 3D computed tomography dataset with 8 labelled landmarks from different types of anatomy such as bone, kidney and vessel bifurcations.

(ii) Instance detection and counting. A number of medical image applications deal with the problem of localizing (potentially a large amount of) visible instances of an object category, which often serves as a key step of frequently occurring quantification for large-scale statistical data analysis [133], computational biology [134] and health informatics [135]. For example, Chen et al. presented an automatic immune cell counting method on digitally scanned images of immunohistochemistry stained slides based on a patch-based deep learning framework. Ronneberger et al. proposed a U-Net deep architecture for cell detection and shape measurements in medical image data [90, 133] by introducing an encoder-decoder structure and a skip connection feature refinement scheme, which inspired a number of studies and applications. Xie et al. proposed to regress a cell spatial density map using a fully convolutional regression network for microscopy cell counting and detection [136]. Sirinukunwattana et al. proposed a spatially constrained CNN to perform nucleus detection in stained histopathology images of cancerous tissue by regressing the likelihood of a pixel being the center of a nucleus. Prentavsic et al. deployed a CNN to detect and count exudates in color fundus photographs for early diagnosis of diabetic retinopathy [137].

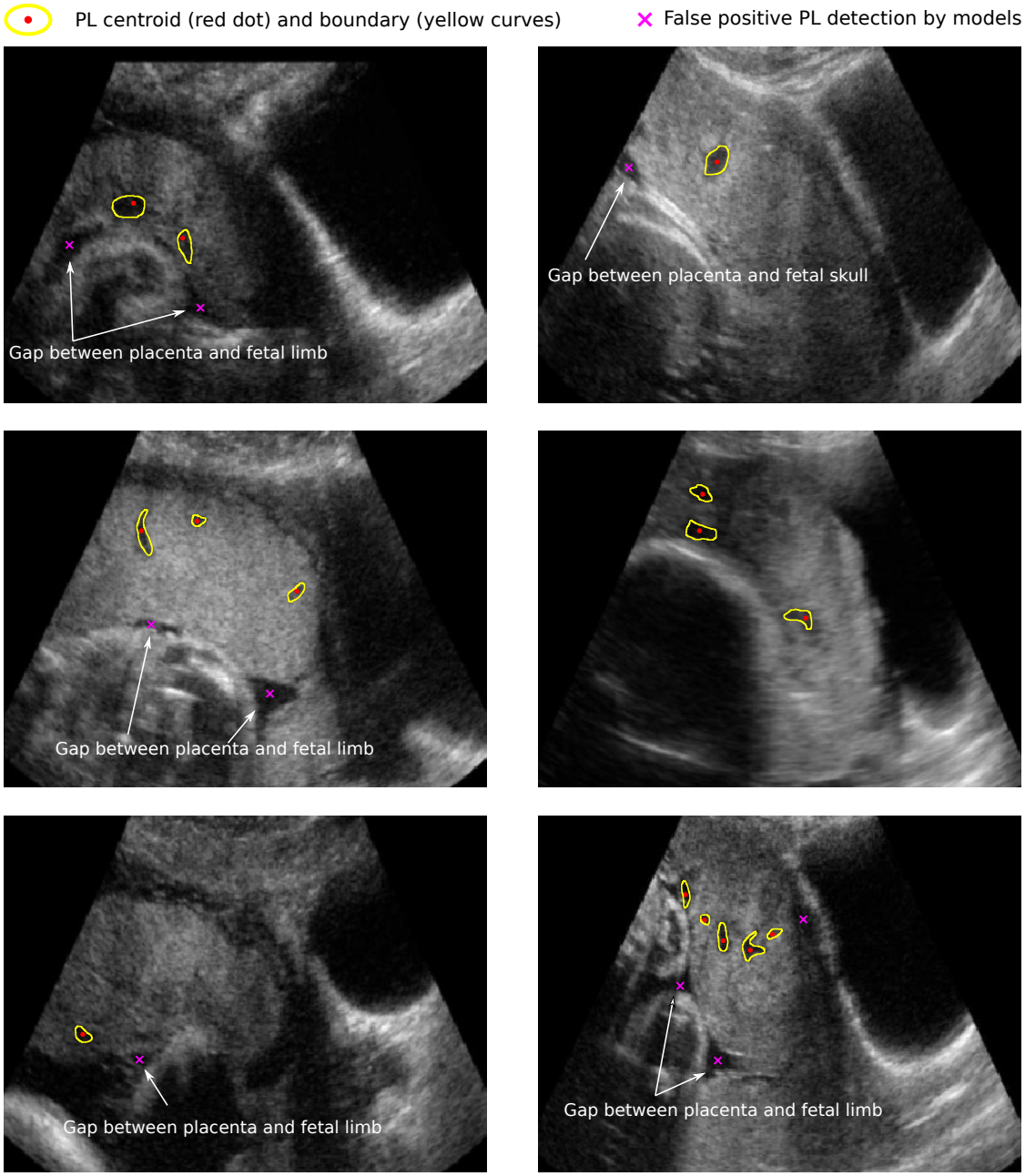


Figure 5.1: Illustration of PL centroids (red dots), PL shapes (yellow curves) and false positive PL predictions (crosses) made by deep learning models. Most of the PL-like false positives are gaps formed by the placenta and fetal limbs or skull.

Localizing placental lacunae (PL) in ultrasound images is a challenging task that lies at the intersection of the two types of problems above. On the one hand, as with anatomical landmark detection tasks, identifying PL relies on both local features and global contextual cues. In particular, the global context can help filter out *PL-like* false positives that resemble PL with respect to sizes and

shapes but differ in the sense that they do not fall inside the placenta, as shown by the crosses in Figure 5.1. On the other hand, much like instance detection tasks, the number of PL instances vary across placental ultrasound images. Taking these characteristics into consideration, the remainder of this chapter presents an automated PL localization method with the following contributions. First, an automated approach is presented that generates confidence maps from expert dot annotations for subsequent training, as an alternative to detailed yet expensive manual segmentation of PL. This method harnesses over-segmentation techniques to generate Gaussian-like confidence maps centered at PL by exploiting local information such as size, shape, and texture of PL. Second, three layer aggregation structures, namely deep supervision (DS), feature pyramid network (FPN) and iterative deep aggregation (IDA), are investigated to merge multi-scale knowledge learned by CNNs. An IDA-based fully convolutional network (FCN) is then proposed for PL localization. Third, a unified evaluation metric based on a normalized Euclidean distance between a PL prediction and the reference location is introduced to measure PL localization performance. Furthermore, a detailed error analysis reveals the strengths and weaknesses of the models.

5.2 From Dot Annotation to Confidence Map

Detailed human labelling, such as manual segmentation of anatomical structures, is sometimes too expensive to carry out in large-scale medical image analysis studies. This chapter investigates a *weak* way of annotating PL, which is referred to as *dot annotation* [136]. The annotation protocol requires that the centroid of the observed PL is pinpointed by the annotator and the spatial coordinates stored. As shown in Figure 5.1, dot annotations pinpoint the most reasonable locations of PL. Learning these coordinates directly in a top-down manner would generally require computational complexity proportional to the number of PL in the image. Instead, an efficient bottom-up approach is presented in this chapter that dissociates runtime complexity from the number of PL by generating a confidence map for each image,

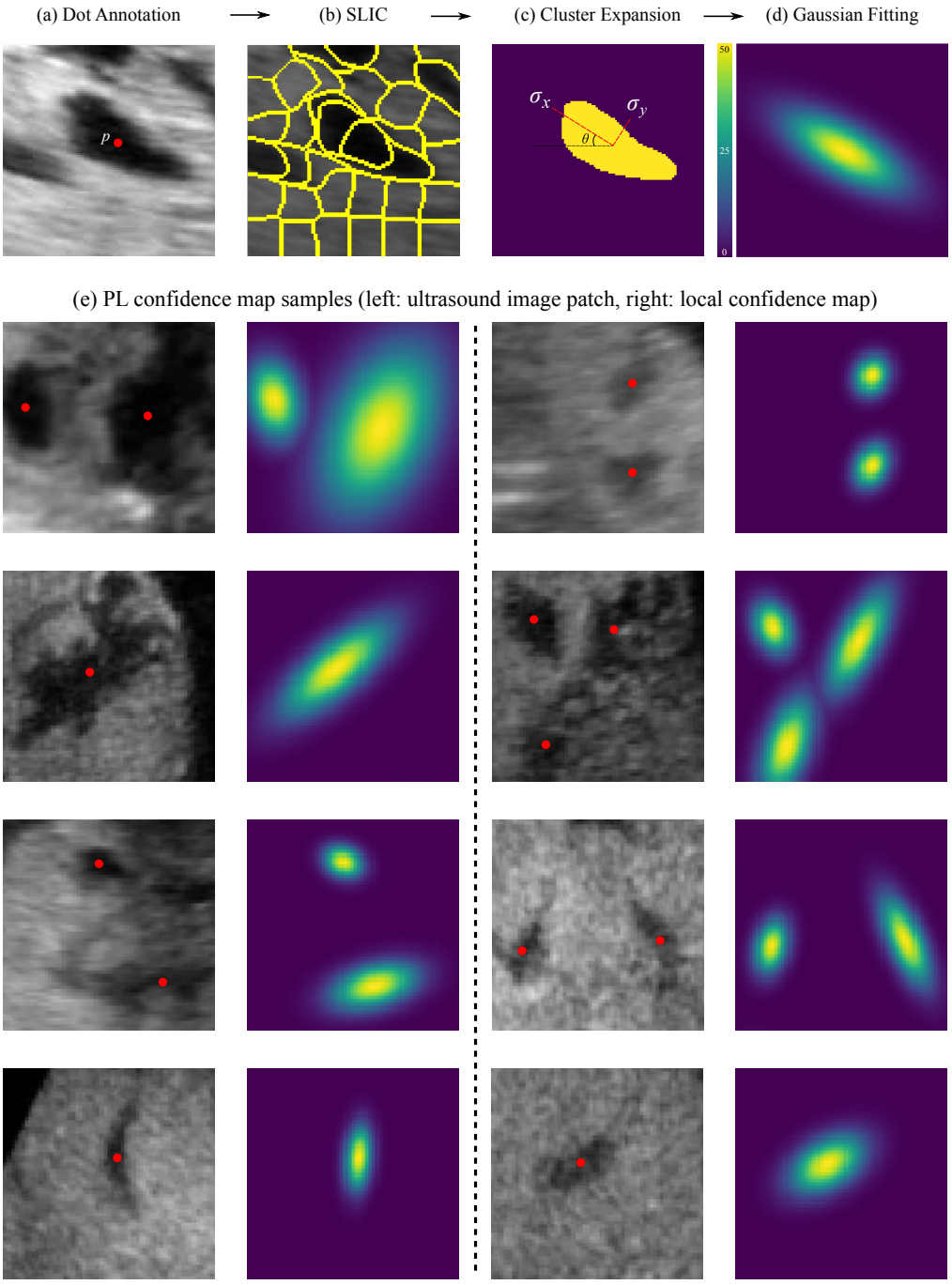


Figure 5.2: The analysis pipeline that generates a local confidence map around a PL given only a dot annotation. (a) a cropped PL image with the dot annotation in the center, (b) the SLIC over-segmentation of the region, (c) a cluster expansion algorithm yielding a binary mask around the dot annotation, (d) a local confidence map is generated by fitting a Gaussian function at the labelled PL, (e) PL confidence map samples.

encoding the belief that PL would occur at each pixel location. Intuitively, dot annotations correspond to peaks in confidence maps.

Let $\mathbf{S}$ denote a confidence map for a placental ultrasound image with a labelled PL at the reference position $\mathbf{p_i} = (p_i^x, p_i^y)$. Previous map generation approaches tend to fit an isotropic Gaussian function centered at $\mathbf{p_i}$ [80, 136], which is given by:

$$\mathbf{S}(\mathbf{l}) = A \exp\left(-\frac{||\mathbf{l} - \mathbf{p_i}||_2^2}{\sigma^2}\right) \tag{5.1}$$

where $\mathbf{l} = (l^x, l^y)$ denotes an arbitrary location in $\mathbf{S}$, σ is a falloff parameter that controls kernel spread and A is a rescaling factor. However, this is a sub-optimal way of generating PL confidence maps by ignoring the local information of PL.

By contrast, this chapter presents an alternative way of generating PL confidence maps by considering the size, shape, and texture in order to improve localization performance. For each labelled PL, a local patch is first cropped, centering at the dot annotation location $\mathbf{p_i}$, as shown in Figure 5.2a. Then the simple linear iterative clustering (SLIC) algorithm is applied on the patch to cluster pixels that are close to each other in a 3-D space spanned by pixel intensity and spatial coordinates [138], as shown in Figure 5.2b. The cluster number is empirically set to 40, with a compactness parameter of 0.1 that balances pixel intensity proximity and space proximity. A simple cluster expansion is then performed by recursively grouping adjacent clusters of similar average pixel intensity via depth-first searching. Starting from the central cluster where the dot annotation lies, the searching traverses neighbour clusters and groups clusters within a difference of average pixel intensity of 3 until there is no group expansion.. The resulting grouped clusters form a binary mask, as shown in Figure 5.2c. In the final step (Figure 5.2d), a 2D Gaussian function is fit at $\mathbf{p_i}$, whose covariance matrix Σ is determined by the ellipse that has the same normalized second central moment as the binary mask in Figure 5.2c, such that the eigenvalues of Σ are the lengths of the major (σ_X) and minor (σ_Y) axes of the ellipse, scaled by a factor $\lambda = 1/3$ in order to control its spread. This results in a 2D Gaussian function in the form:

$$\mathbf{S}(l_x, l_y) = A \exp\left(-\left(a_S(l^x - p_i^x)^2 + 2b_S(l^x - p_i^x)(l^y - p_i^y) + c_S(l^y - p_i^y)^2\right)\right) \tag{5.2}$$

where a_S, b_S, c_S are defined as:

$$
\begin{aligned}
a_S &= \frac{\cos^2\theta}{2(\lambda\sigma_X)^2} + \frac{\sin^2\theta}{2(\lambda\sigma_Y)^2} \\
b_S &= -\frac{\sin 2\theta}{4(\lambda\sigma_X)^2} + \frac{\sin 2\theta}{4(\lambda\sigma_Y)^2} \\
c_S &= \frac{\sin^2\theta}{2(\lambda\sigma_X)^2} + \frac{\cos^2\theta}{2(\lambda\sigma_Y)^2}
\end{aligned}
\tag{5.3}
$$

where θ refers to the angle between the x-axis and the major axis of the fitted ellipse (Figure 5.2c). The rescale factor A of the Gaussian function is set to 50 as suggested in [136]. By computing Equation 5.2 for all PL within an ultrasound image, a smoothed confidence map $\mathbf{S}$ is generated. Where two or more Gaussian functions spatially overlap with each other, the pixel-wise *maximum* of the overlapping regions is used. Figure 5.2e displays more confidence map samples generated from the expert dot annotations (red dots).

5.3 Deep Layer Aggregation for Representation Learning

This section focuses on deep architectures that promote representation learning across different image scales. For convenience, layers that yield the same feature resolution are put into the same *stage*. FCN's natural pyramidal feature hierarchy enables aggregations of both spatial (i.e. where) and semantic (i.e. what) information from shallower stages to deeper ones. To achieve more accurate spatial inference of PL, whose size is always much smaller than the placenta itself, this chapter proposes to build up *non-linear* pathways that explicitly aggregate multi-scale semantics and resolutions. Specifically, two generic FCN architectures that have been widely used in medical image analysis are investigated: (i) a downsampling network (DN) and (ii) a U-shape network (UN). As shown in Figure 5.3a and Figure 5.4a, DN sequentially down-samples stages, leading to semantically richer but spatially coarser features [139]. UN follows an encoder-decoder architecture, with the encoder part being a DN and the decoder part an up-sampling network that gradually restores resolution via 2×2 transposed convolution [90].

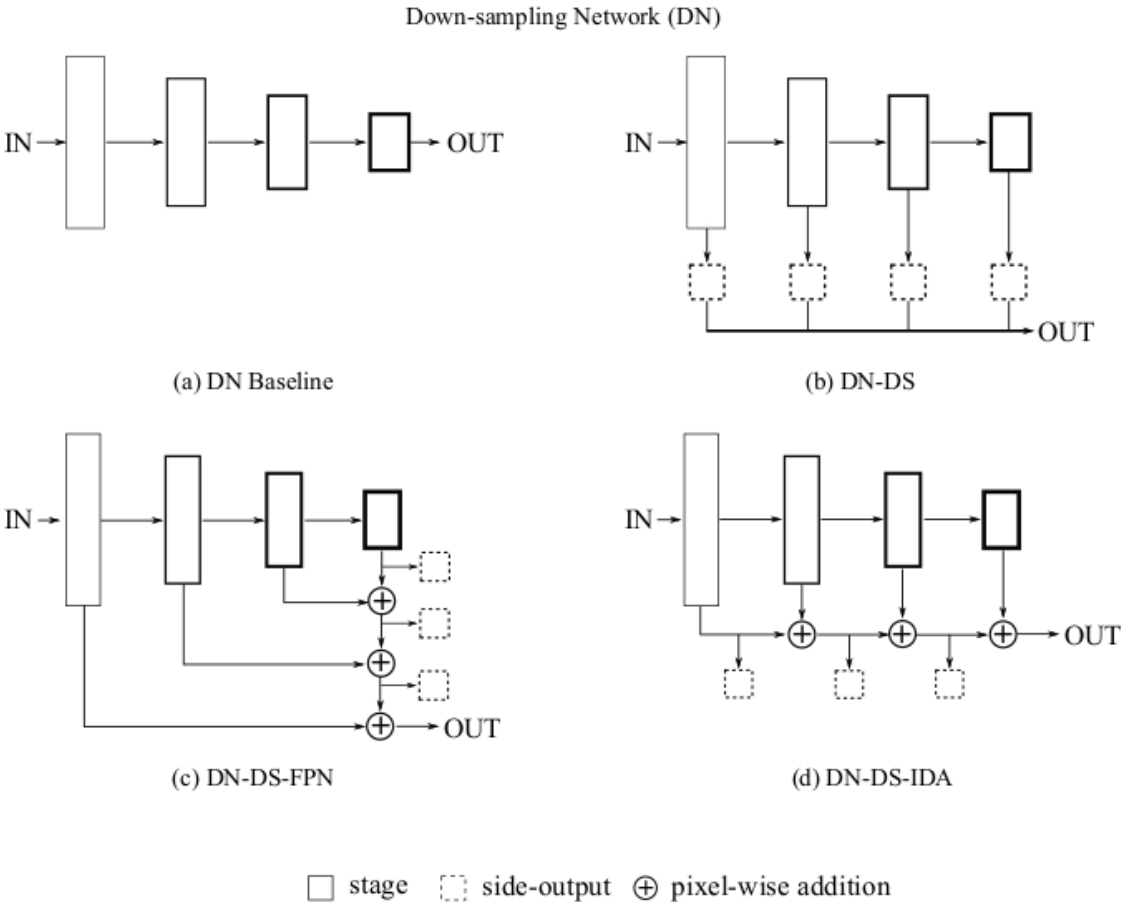


Figure 5.3: Down-sampling networks (DN) with three layer aggregation structures: (a) deep supervision (DN-DS), (b) deeply supervised feature pyramid network (DN-DS-FPN), and (c) deeply supervised iterative deep aggregation (DN-DS-IDA). Intuitively, the box size is proportional to the spatial resolution and the box linewidth to the feature channel number.

Three layer aggregation approaches are considered here: (i) deep supervision (DS) [140]; (ii) feature pyramid network (FPN) [141]; and (iii) iterative deep aggregation (IDA) [142]. As shown in Figure 5.3b and Figure 5.4b, DS concatenates intermediate side-outputs to make the final prediction. Here a side-output is a prediction made by the output of a stage. FPN intends to enhance semantically stronger features (from deeper stages) with weaker ones (from shallower stages) via skip connection and *linear* pixel-wise addition. In FPN, the shallowest stage will be aggregated last. IDA, on the other hand, starts from the shallowest stage and iteratively merges deeper ones. All feature channel mismatches are resolved by 1×1 convolution and resolution mismatches by bilinear upsampling.

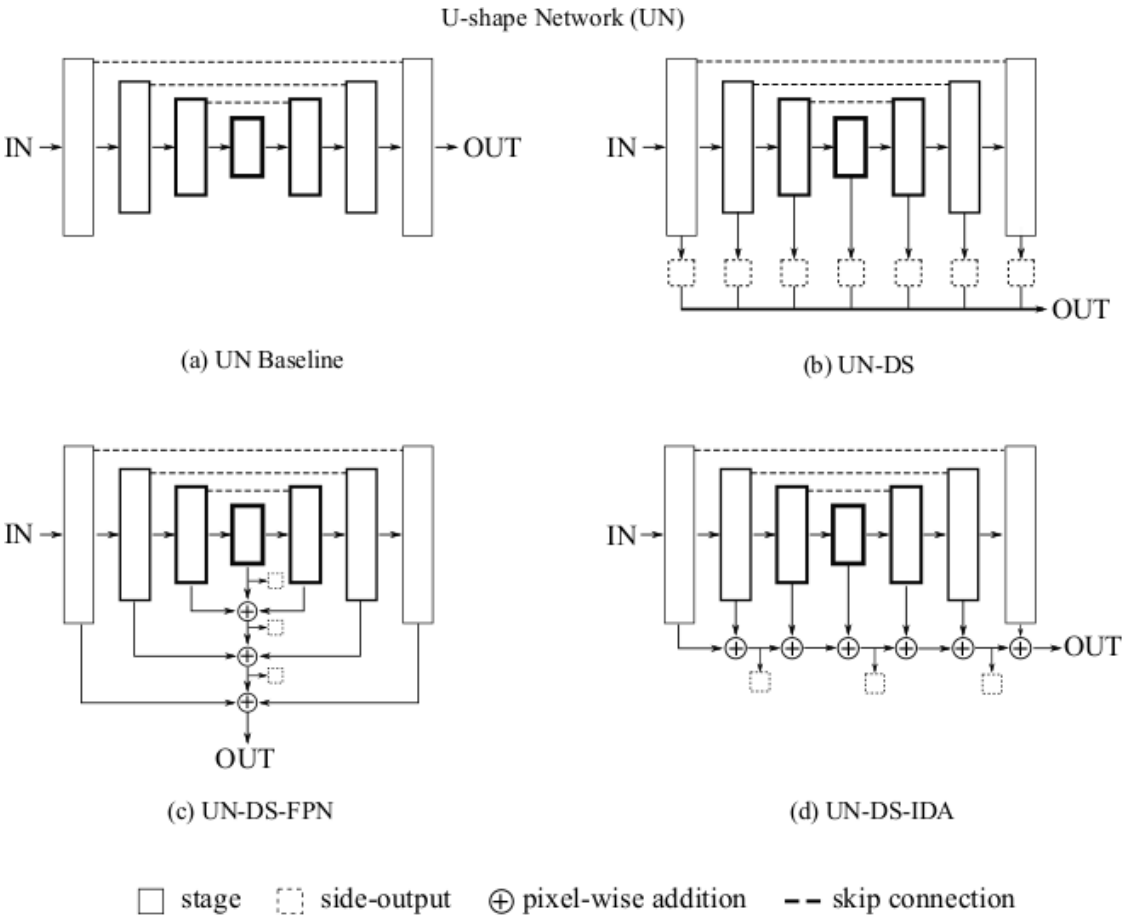


Figure 5.4: U-shape networks (UN) with three layer aggregation structures: (a) deep supervision (UN-DS), (b) deeply supervised feature pyramid network (UN-DS-FPN), and (c) deeply supervised iterative deep aggregation (UN-DS-IDA). Intuitively, the box size is proportional to the spatial resolution and the box linewidth to the feature channel number.

To achieve accurate PL localization, two layer aggregation structures are proposed, namely deeply supervised feature pyramid network (DS-FPN, Figure 5.3c and Figure 5.4c) and deeply supervised iterative deep aggregation (DS-IDA, Figure 5.3d and Figure 5.4d). For a set of input feature maps $\{\mathbf{x_i}\}$, non-linear pixel-wise additions are performed in both structures:

$$f(\mathbf{x_i}) = \sigma(\mathrm{BN}(\sum_i \mathbf{w_i}\mathbf{x_i})) \tag{5.4}$$

where σ_f is a non-linearity (e.g. ReLU), BN is a batch normalization layer, and $\mathbf{w_i}$ are convolutional weights to be learnt. Side-outputs are produced in both structures to cast additional supervision alongside the aggregation pathways. Intuitively,

DS-FPN focuses more on semantically stronger features from deeper stages while DS-IDA progressively enhances spatially finer features from shallower stages. This comparison allows investigation on the importance of features from shallower versus deeper stages in PL localization. The objective function of DS-FPN and DS-IDA are the same, which is given by

$$\mathcal{L}(\mathbf{W}) = \ell(S_{OUT}, \hat{S}) + \frac{1}{N}\sum_{i=1}^{N} \ell(S_i, \hat{S}) \tag{5.5}$$

Here $\hat{S}$ is the reference confidence map, S_{OUT} is the final output of the model, and $\{S_i\}_{i=1}^{N}$ are N side-outputs. $\mathbf{W}$ represents all the learnable parameters. $\ell(\cdot,\cdot)$ denotes the L-2 loss between the inputs. The final output S_{OUT} encodes PL localization confidence. PL centroid predictions are obtained by performing non-maximum suppression.

5.4 Simplified Object Keypoint Similarity

PL localization requires simultaneous classification and localization. Previous anatomical landmark detection and instance counting tasks evaluate model performance using a number of different metrics, such as errors in millimeters [129, 132] and count error rate [136]. However, these metrics do not measure precision and recall (sensitivity) directly, which tends to give a more informative picture of a model's performance in a detection task such as PL localization [143].

Inspired by the COCO human pose estimation challenge, this section introduces an evaluation metric for PL localization that has not yet been widely used in anatomical landmark detection tasks. Given a reference PL indexed by i, its dot annotation $\mathbf{p_i} = (p_i^x, p_i^y)$ is provided, i.e. coordinates of its centroid. Each PL also has a scale s_i which is defined as the square root of its SLIC cluster area measured in pixels. Accordingly, a simplified object keypoint similarity (SOKS) score for a PL prediction j on a reference PL i is given by:

$$\text{SOKS}(j \to i) = \exp\left(\frac{-d_{ji}^2}{2s_i^2\kappa^2}\right) \tag{5.6}$$

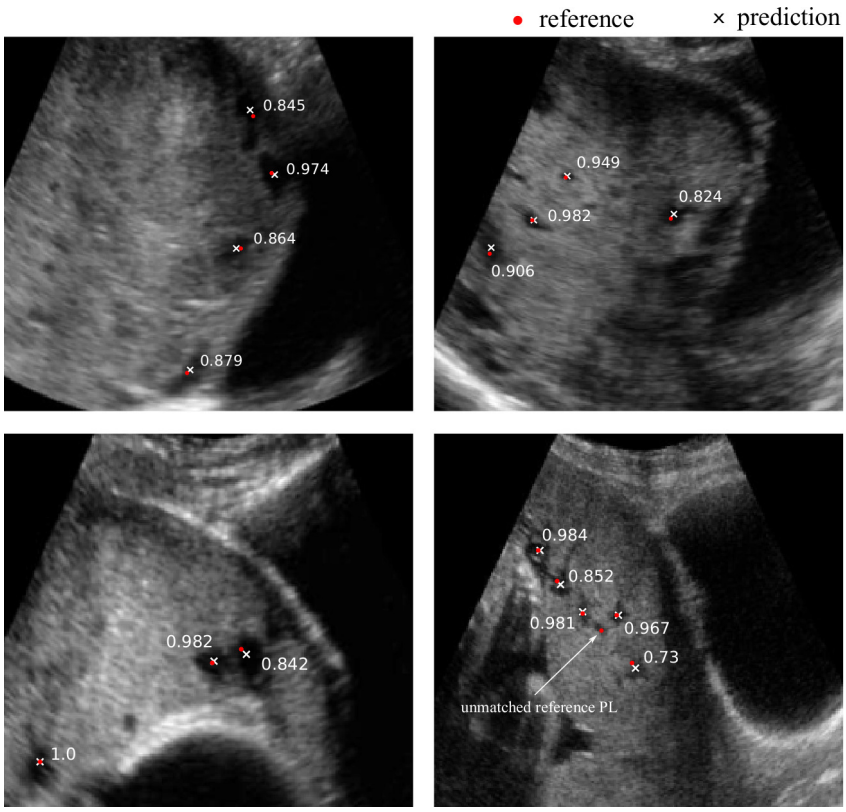


Figure 5.5: Example images illustrating the use of SOKS to score PL localization. In each image, a red dot denotes the dot annotation of a reference PL and a white cross denotes the prediction. The value next to a pair of dot and cross is the SOKS score between them. Note that every PL prediction has an associated SOKS score to its nearest reference PL. However, there can be unmatched reference PL, for example the one in the bottom-right image.

where d_{ji} is the Euclidean distance between prediction j and reference i. The constant κ controls the overall falloff, which is empirically set to 0.424. Please refer to the experiment section (§ 5.5) for details in determining the value of κ. The intuition behind SOKS is that larger tolerance is given to a PL with larger scale s_i. Empirically, SOKS$\geq$ 0.3 yields a perceptually acceptable localization, as shown in Figure 5.5.

Much like the Intersection over Union (IoU) score for bounding-box-based object detection[2], a SOKS score is a measure of how close a PL prediction is to a reference PL. The evaluation starts with matching PL predictions to references. First, each PL prediction is paired with a reference that yields the highest SOKS. In the circumstance where the highest SOKS is smaller than 0.05, the PL prediction is

[2]`http://host.robots.ox.ac.uk/pascal/VOC/`

considered unmatched. Second, any reference PL with more than one paired PL predictions will only match with the prediction that yields the highest SOKS, while the rest PL predictions become unmatched. Before diving into details, the following basic concepts are first presented given a hold $\epsilon \in [0, 1]$:

- **True positive** (TP): a correct PL prediction, with SOKS $\geq \epsilon$
- **False positive** (FP): a wrong PL prediction, with SOKS $< \epsilon$
- **False negative** (FN): a unmatched reference PL
- **Ignored** (IG): a correct PL prediction, with SOKS $\geq \epsilon$, but unmatched
- **Precision**: a model's capability to identify only the relevant instance, which is the ratio of true positives among all predictions:

$$Precision = \frac{TP}{TP + FP} \tag{5.7}$$

- **Recall (sensitivity)**: a model's capability to find all the relevant instances, which is the ratio of true positives among all references:

$$Recall = \frac{TP}{TP + FN} \tag{5.8}$$

In this chapter, a PL localizer is considered good if its precision stays high as its recall increases. Following the common evaluation protocol human pose estimation, the average precision (AP), which is the area under a precision-recall curve, is used for quantitative performance evaluation. For a given precision-recall curve, a 101-point interpolation computes AP by averaging the precision at a set of 101 equally spaced recall levels $\{0 : 0.01 : 1\}$:

$$AP = \frac{1}{101} \sum_{r \in \{0:0.01:1\}} p(r) \tag{5.9}$$

with

$$p(r) = \max_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r}) \tag{5.10}$$

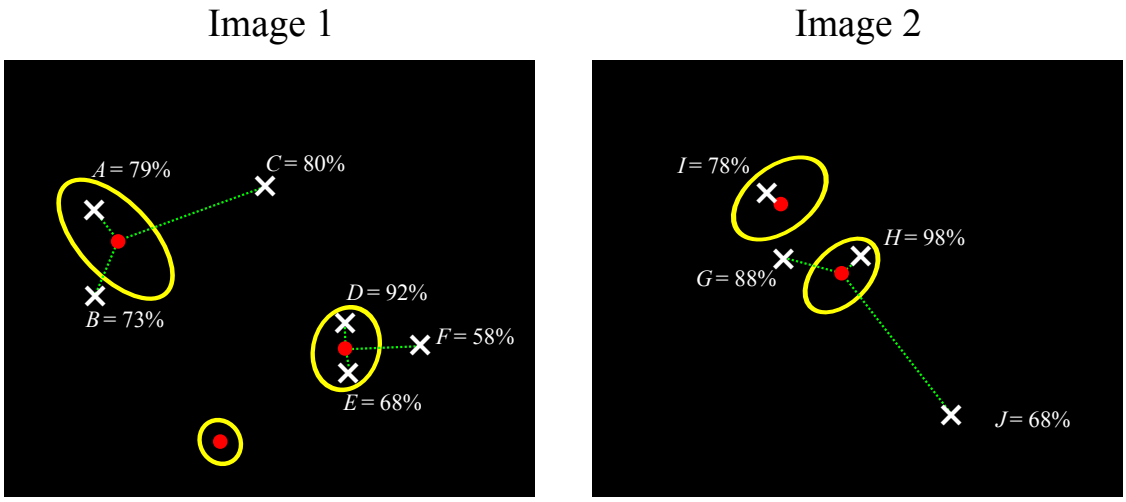


Figure 5.6: A simulated example demonstrating the process of computing average precision given a set of detections and reference labels.

Table 5.1: Model Predictions in Figure 5.6 with SOKS Threshold set to 0.3 and 0.75

Image	Prediction	Confidence	SOKS (w.r.t. nearest PL)	Label @ 0.3	Label @ 0.75
1	A	79%	0.784	TP	TP
1	B	73%	0.276	FP	FP
1	C	80%	∼0	FP	FP
1	D	92%	0.641	TP	FP
1	E	68%	0.632	IG	FP
1	F	58%	0.023	FP	FP
2	G	88%	0.052	FP	FP
2	H	98%	0.569	TP	FP
2	I	78%	0.912	TP	TP
2	J	68%	∼0	FP	FP

This chapter uses AP_ϵ to measure the performance of a PL localizer, which denotes the AP by thresholding SOKS at ϵ. To achieve a high score of AP_ϵ, a model needs to have high precision at *all* levels of recall (or sensitivity), which is practically difficult in PL localization. In particular, $AP_{0.3}$, $AP_{0.5}$, $AP_{0.75}$, and mAP are reported in § 5.5, where mAP is the mean of $\{AP_\epsilon\}$ for $\epsilon \in \{0.3 : 0.05 : 0.95\}$, measuring the overall localization performance at different SOKS levels.

To better demonstrate the evaluation process, a simulated example dataset containing two images is displayed in Figure 5.6. Details of the simulated predictions and references are listed in Table 5.1. The resulting AP calculation process is presented in Table 5.2. Suppose a PL localizer has made its prediction over these

Table 5.2: Computing Precision and Recall for Predictions in Figure 5.6 with SOKS Threshold Set to 0.3

Image	Prediction	Confidence	TP	FP	Acc TP	Acc FP	Precision	Recall
2	H	98%	1	0	1	0	1	0.2
1	D	92%	1	0	2	0	1	0.4
2	G	88%	0	1	2	1	0.667	0.4
1	C	80%	0	1	2	2	0.5	0.4
1	A	79%	1	0	3	2	0.6	0.6
2	I	78%	1	0	4	2	0.667	0.8
1	B	73%	0	1	4	3	0.571	0.8
2	J	68%	0	1	4	4	0.5	0.8
1	F	58%	0	1	4	5	0.444	0.8
							$AP_{0.3}$=0.670	

two images, which are marked as white crosses ($A \sim J$) with the corresponding prediction confidence scores. Five reference PLs are marked as red dots with different scales, i.e. yellow contours of different sizes. Furthermore, Table 5.1 shows the predictions with the corresponding confidence scores and SOKS scores with respect to their matched references (marked with green dotted lines in Figure 5.6). The last two columns identify the predictions as TP, FP or IG at two SOKS levels: $\epsilon = 0.3$ and $\epsilon = 0.75$. Note that IG refers to a ignored prediction, which is introduced in the COCO detection challenge[3] to prevent identifying good but unmatched predictions as false positives (such as prediction E in Figure 5.6). Table 5.2 takes $\epsilon = 0.3$ as an example. The precision and recall are computed using Equation 5.7 and Equation 5.8 by first sorting predictions according to their confidence scores from high to low, as shown by the last two columns in Table 5.2, followed computing AP using Equation 5.9.

5.5 Experiments & Discussions

This section describes experiments to evaluate predictions made by various PL localizers, followed by discussions on several design choices. First, the dataset used in all experiments are introduced in § 5.5.1, along with how to determine the value of κ presented in Equation 5.6. Implementation details such as hyper-parameter

[3] `http://cocodataset.org/#detection-eval`

tuning are listed in § 5.5.2. Performance analysis is presented in § 5.5.3, followed by discussions on the suitability of the proposed approach for clinical applications.

5.5.1 Dataset

3,440 image planes from 34 placental ultrasound volumes were annotated for PL localization (see § 3.3), 23 of which were complicated by PASD. Due to the limited amount of data available for this task, a subject-level 10-fold nested cross-validation[4] was used for performance evaluation. In each fold, test data consisting of image planes from 3 - 4 volumes were held out while images from the remaining volumes were used for training and validation. In other words, all the image data would be tested exactly once in some fold, which compensated for the data shortage to some extent.

As defined in Equation 5.6, SOKS is essentially a Gaussian function, where s_i is the PL scale and κ is a constant that controls the overall falloff. By definition, SOKS $\in (0, 1]$. A perfect prediction has a SOKS score of 1 and predictions that are off by a few standard deviations will have a SOKS score close to 0. For this task, the value of κ is determined by measuring PL manual annotation standard deviation σ_κ with respect to scale s_i using 50 redundantly annotated images in a held-out validation set. Specifically, κ is estimated by

$$\kappa = 2\sigma_\kappa \tag{5.11}$$

with

$$\sigma_\kappa^2 = \mathbb{E}\left[\frac{d_{ji}^2}{s_i^2}\right] \tag{5.12}$$

The estimated κ is 0.424. With this setting, an SOKS score would take on values of 0.88, 0.61, and 0.32 at one, two, and three standard deviations of d_{ji}/s_i.

[4]For more details on nested cross-validation, please refer to § 6.3.2.

Table 5.3: mAP of different PL localizers on the test set via 10-fold cross validation. All results (%) are in the format of *median [first, third quartile]*. Models are named in the format of *A*-*B*, where *A* is its generic architecture (DN or UN) and *B* its layer aggregation structure (DS or DS-FPN or DS-IDA)

Model	Backbone	mAP
DN-DS	ResNet-18	22.8 [20.2, 26.5]
DN-DS-FPN		28.7 [22.0, 30.0]
DN-DS-IDA		29.7 [25.3, 34.8]
UN-DS	VGG-16	32.6 [24.1, 37.5]
UN-DS-FPN		32.2 [28.4, 37.4]
UN-DS-IDA		**35.7** [28.4, 40.7]

5.5.2 Implementation Details

ResNet-18 and VGG-16 were chosen as model backbones for DN and UN respectively [83, 84]. The first 7×7 convolutional layer and the max-pooling layer from ResNet-18 were removed such that all models contained three down-sampling operations. All models were trained end-to-end using the Adam optimizer [74]. ImageNet pre-trained parameters of ResNet-18 and VGG-16 were loaded for DN as well as the encoder part of UN. The decoder part of UN was initialized by sampling from a Gaussian distribution $\mathcal{N}(0, \sqrt{2/n})$, where n is the number of trainable parameters for each layer. The inputs were normalized to have zero mean and unit variance and resized to have the dimension of $384 \times 384 \times 3$. The horizontal flip was used for data augmentation. The hyper-parameters were: mini-batch size 8; weight decay 0.0005; initial learning rate 0.0001. All models reached convergence after 20 epochs. All experiments were implemented using the deep learning library PyTorch. A 10-fold training took around 30 hours on a 12GB NVIDIA GTX 1080Ti graphic card.

5.5.3 Performance Analysis

Performances of different PL localizers are given in Table 5.3 and Table 5.4. The median, first, and third quartile of 10-fold results are presented. The proposed UN-DS-IDA surpasses all other PL localizers in all AP metrics. Two-tailed paired t-tests show that mAP from UN-DS-IDA is significantly higher than those from the rest PL localizers (with p-value < 0.001).

Table 5.4: AP_ϵ of different PL localizers on the test set via 10-fold cross validation. All results (%) are in the format of *median [first, third quartile]*. AP_ϵ at three SOKS thresholds ($\epsilon \in \{0.3, 0.5, 0.75\}$) are reported. Models are named in the format of *A-B*, where *A* is its generic architecture (DN or UN) and *B* its layer aggregation structure (DS or DS-FPN or DS-IDA)

Model	Backbone	$AP_{0.3}$	$AP_{0.5}$	$AP_{0.75}$
DN-DS	ResNet-18	33.0 [30.3, 36.3]	29.6 [27.1, 33.8]	19.7 [16.9, 23.9]
DN-DS-FPN		42.7 [34.5, 43.6]	37.5 [30.6, 39.3]	24.0 [16.9, 25.7]
DN-DS-IDA		38.6 [33.8, 46.0]	36.3 [31.0, 43.9]	28.5 [24.1, 32.5]
UN-DS	VGG-16	41.4 [35.2, 47.2]	39.6 [31.4, 44.1]	31.3 [22.0, 36.5]
UN-DS-FPN		42.3 [39.7, 46.0]	40.2 [35.7, 44.1]	31.0 [24.8, 36.8]
UN-DS-IDA		**44.7** [40.9, 50.1]	**42.3** [36.5, 48.5]	**35.3** [26.4, 37.8]

Unless specified otherwise, the rest of this chapter presents analysis on only one fold, which was randomly chosen from the 10-fold split, in order to avoid reporting similar results many times. For simplicity, this fold is referred to as *fold-1*. Figure 5.7 displays the precision-recall curves of the proposed DS-IDA and DS-FPN models on the test set of fold-1 at four different SOKS levels: $\epsilon \in \{0.3, 0.5, 0.7, 0.9\}$. UN-DS-IDA generally outperforms the rest models in terms of recall (sensitivity), i.e. finding more relevant PL instances, which is desirable in clinical applications. More test samples from fold-1 made by UN-DS-IDA are displayed in Figure 5.8.

To further investigate the effectiveness of DS-IDA, the localization performance of side-outputs are probed. Let the side-outputs (from left to right) in DN-DS and UN-DS be $\{S_i\}_{i=1}^4$ and $\{\tilde{S}_i\}_{i=1}^7$ respectively. In the 10-fold cross validation experiment, the mAP score (median) of S_1, S_2, $\tilde{S}_1$, and $\tilde{S}_2$ are 0. Starting from S_3 and $\tilde{S}_3$, mAP scores increase as expected. Consequently, it is argued that features from shallower stages are not effectively aggregated via either concatenation (DS) or skip connection (FPN). Moreover, since features from the shallowest stage in DS-FPN are aggregated last, there is limited room for convolutional kernels to adapt throughout training. On the contrary, DS-IDA structure progressively aggregates features from shallower stages. The experimental results indicate that features from shallower stages can indeed contribute to PL localization effectively with the proposed DS-IDA structure. One reasonable explanation is that down-sampling

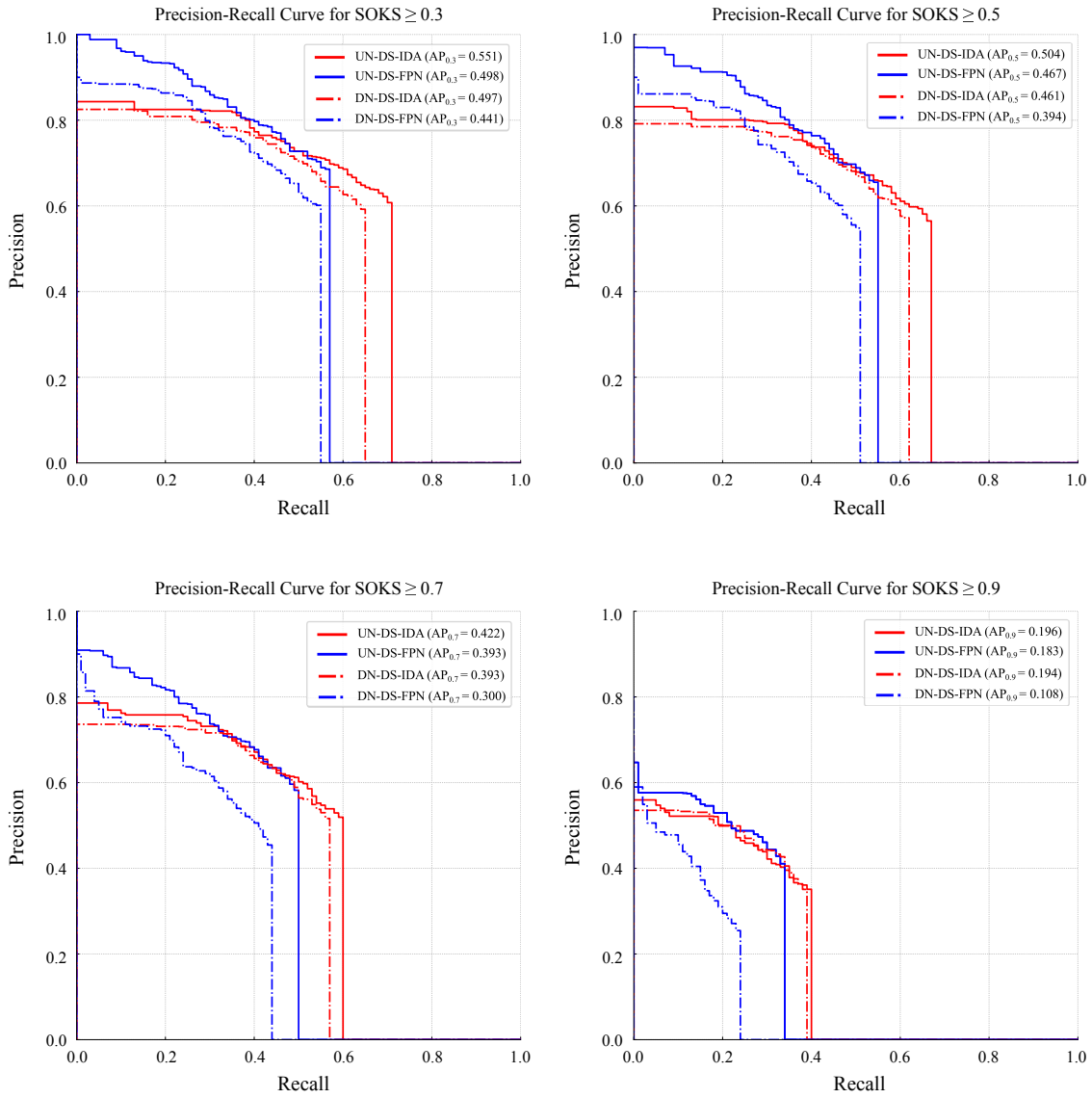


Figure 5.7: Precision-recall curves on the test set of fold-1 at four SOKS levels: $\epsilon \in \{0.3, 0.5, 0.7, 0.9\}$.

operation would lose certain PL-related spatial information. Aggregating shallower features compensate for such loss to some extent.

In the previous section (§ 5.2), an SLIC-based approach are presented that generates confidence maps by taking into account the size, shape, and texture of PL, compared with an isotropic approach that fits Gaussian at each dot annotation with a fixed falloff. To validate this design choice, the isotropic Gaussian approach was also considered. Specifically, an isotropic Gaussian function was fit at each PL to generate confidence maps following Equation 5.1. The best localization

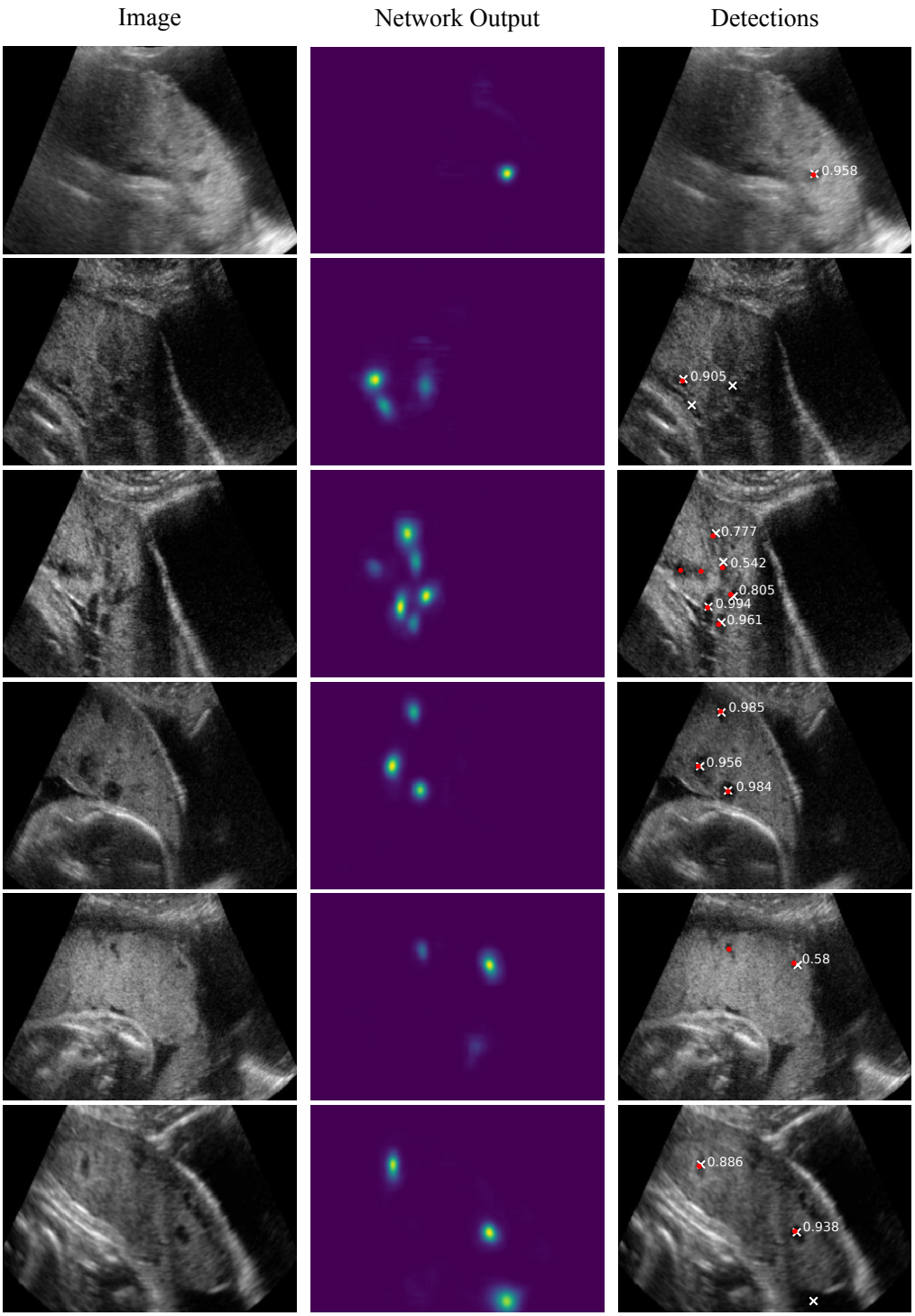


Figure 5.8: Test samples of UN-DS-IDA. The left column presents the input image samples. The middle column presents the network outputs. The right column shows predictions (white cross) and references (red dot), along with the associated SOKS scores for matched pairs.

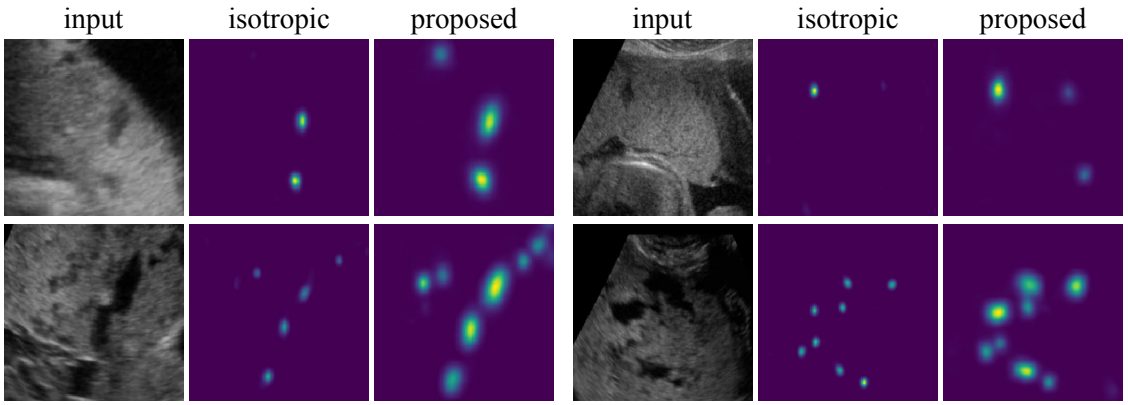


Figure 5.9: Example results showing two model outputs, trained using isotropic Gaussian confidence map and the proposed SLIC-based confidence map respectively.

performance was achieved by a UN-DS-IDA model at $\sigma = 5$ with mAP= 29.9%, outperformed by the proposed SLIC-based approach. This is due to the fact that the size and shape of PL are variable. There is no optimal σ that would achieve good localization for PL with variable sizes and shapes. The SLIC-based approach uses local information in an adaptive manner, which makes it well-suited to PL localization. Moreover, the proposed approach leads to better visualization that learns the size and shape automatically, as shown in Figure 5.9.

5.6 Error Analysis

Inspired by the method proposed by Ronchi et al. to analyze the impact of errors in human pose estimation algorithms [123], this section performs a detailed error analysis on the proposed top 3 PL localizers trained on fold-1: UN-DS-IDA, UN-DS-FPN and UN-DS. Their localization performances are presented in Table 5.5. In particular, two main classes of errors are defined and characterized: localization errors and background errors. Analysis focuses on how these errors are influenced by PL instance attributes such as instance number and examines their impacts on an algorithm performance, with the hope that the analysis framework would provide some insight for understanding the behaviours of the proposed algorithms and measure their strengths and weaknesses for PL localization.

Table 5.5: PL Localizers' Performance on the test set of fold-1

Model	mAP	$AP_{0.3}$	$AP_{0.5}$	$AP_{0.7}$
UN-DS-IDA	0.410	0.551	0.504	0.422
UN-DS-FPN	0.381	0.498	0.467	0.393
UN-DS	0.386	0.520	0.478	0.381

5.6.1 Error Type

Based on the taxonomy of errors in [123], an error in PL localization can be classified as: (i) a localization error (Figure 5.10a), due to the poor localization of a PL prediction to its matched reference; (ii) a background error (Figure 5.10b), due to failed matching. The definition and impact on the performance of PL localizers are presented for each error type.

(i) Localization Errors

A localization error occurs when the SOKS score between a PL prediction and its matched reference PL is lower than a pre-defined threshold, which usually results from the fact that PL centroids are difficult to localize due to the variable shapes and sizes of PLs. Two types of localization errors are defined, as shown in Figure 5.10a, according to the SOKS score (Equation 5.6) between a PL prediction j and its matched reference PL i.

- *Jitter*: a small error around the reference PL, which leads to:

$$0.3 \leq \text{SOKS}(j \rightarrow i) < 0.7 \tag{5.13}$$

- *Miss*: a large error around the reference PL, which leads to:

$$0.05 \leq \text{SOKS}(j \rightarrow i) < 0.3 \tag{5.14}$$

The thresholds are chosen based on observations that SOKS ≥ 0.3 generally yields a perceptually acceptable localization; SOKS ≥ 0.7 would lead to an *acceptable* prediction that would cause significant disagreement even among human annotators in estimating the PL centroid; SOKS < 0.05 indicates that the prediction is too

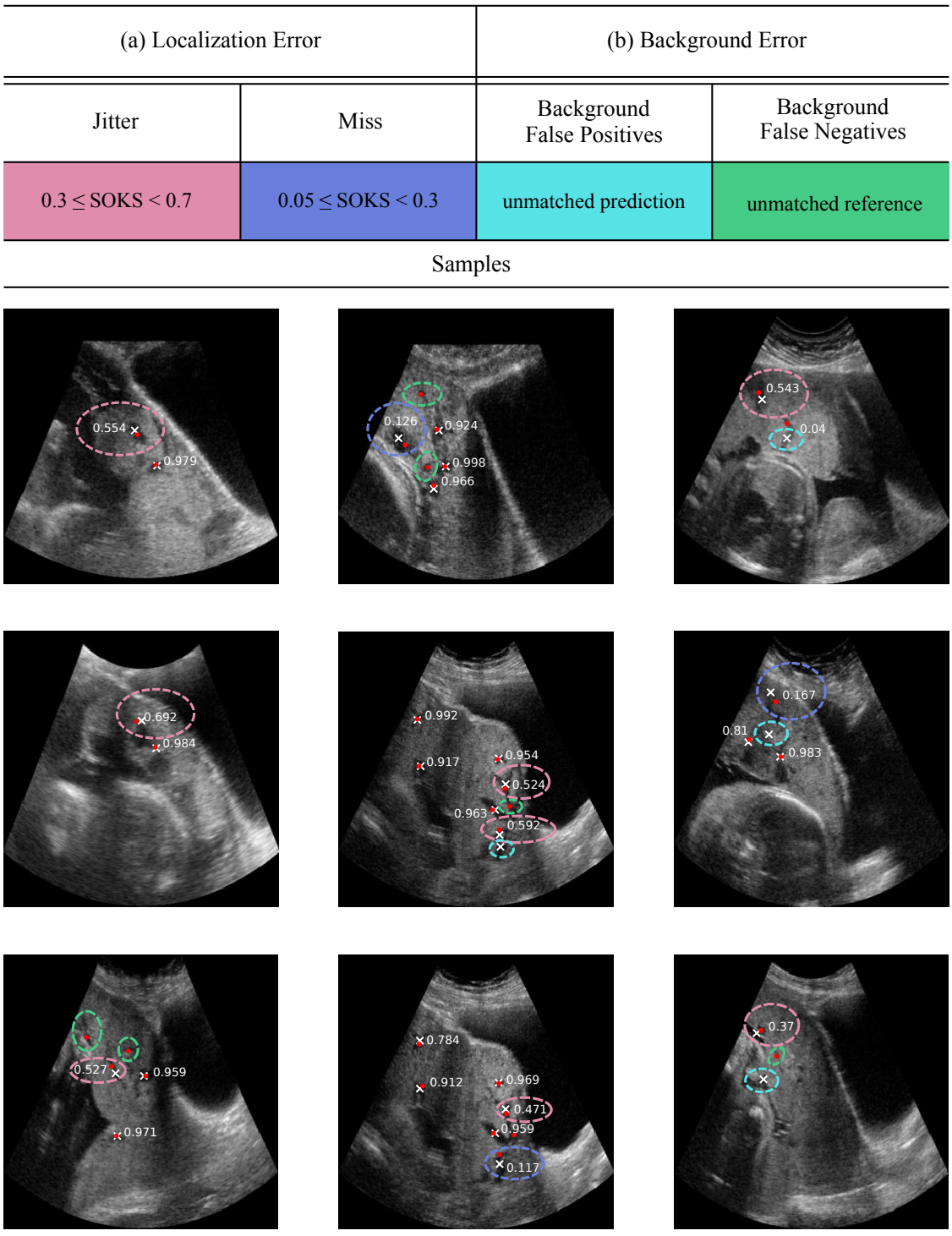


Figure 5.10: Two main classes of errors are defined based on SOKS scores and the matching scheme introduced in § 5.4. A localization error can be further categorized as a jitter or a miss, while a background error is either a false positive or a false negative. Testing samples are displayed to illustrate errors of different types via color encodings. Best viewed in color.

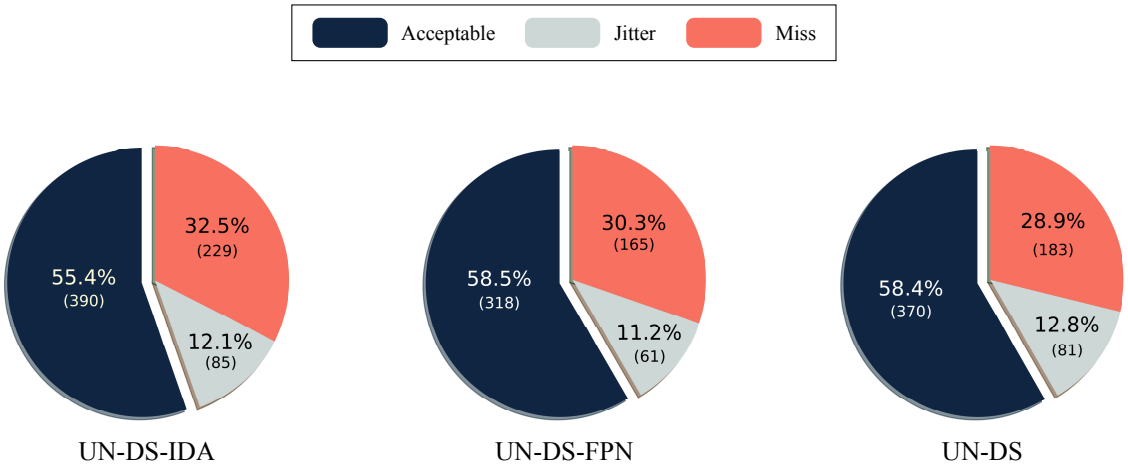


Figure 5.11: The frequency of localization errors over all matched PL predictions.

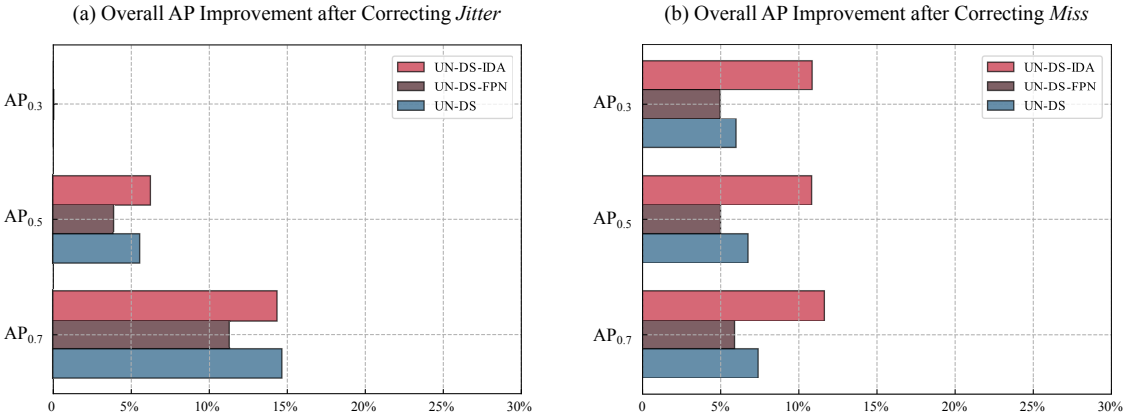


Figure 5.12: Overall AP improvements obtained after correcting localization errors; evaluated at SOKS levels of {0.3, 0.5, 0.7}.

far away from its nearest reference PL to count as a qualified prediction since an SOKS score is essentially an inversed normalized Euclidean distance. Therefore, all localization errors discussed here refer to PL predictions with SOKS ≥ 0.05. The remaining unqualified predictions (i.e. with SOKS < 0.05) are categorised as background errors, which will be covered later in this subsection.

Note that the evaluation metric AP_ϵ is a function of ϵ, while localization errors are identified using fixed SOKS thresholds. This categorization allows further investigations on the distribution and impact of localization errors for each examined PL localizer. As shown in Figure 5.11, more than 55% of the localization errors made by the three PL localizers are *acceptable*, i.e. with SOKS scores larger than 0.7. UN-DS-IDA made more localization errors than the other two models in

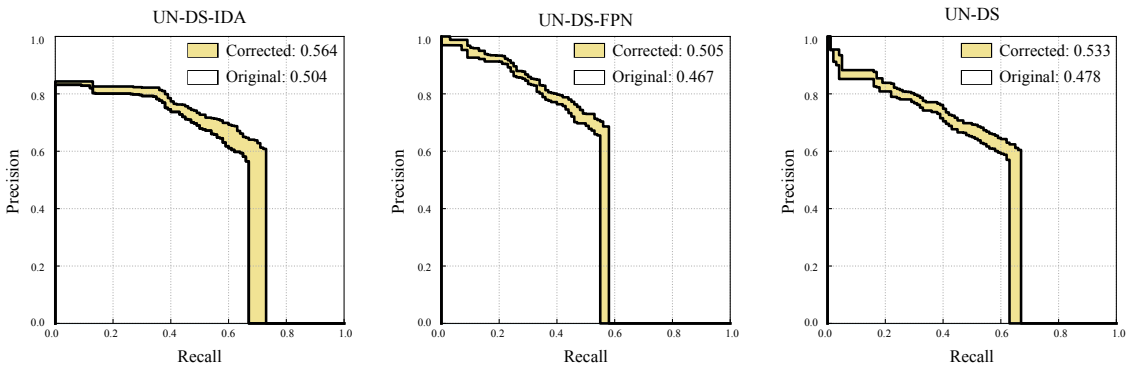


Figure 5.13: Change of precision-recall curves before and after correcting *jitter* errors; evaluated at SOKS level of 0.5.

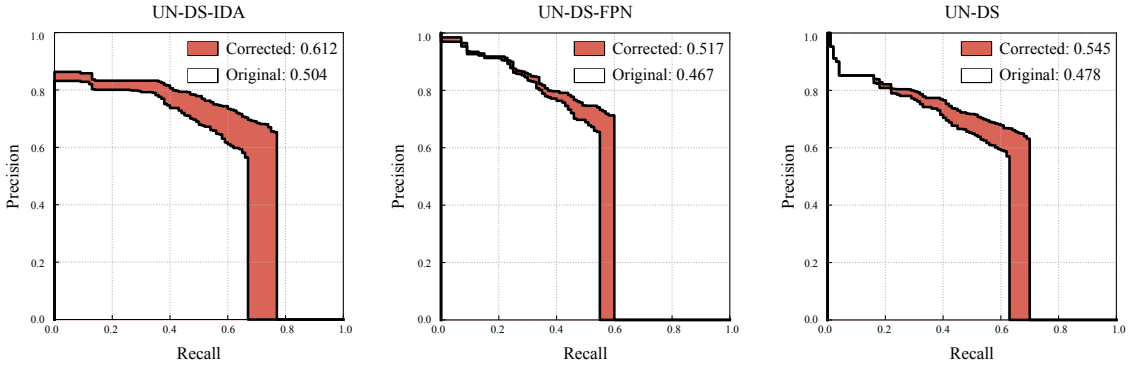


Figure 5.14: Change of precision-recall curves before and after correcting *miss* errors; evaluated at SOKS level of 0.5.

acceptable, jitter, miss categories while still outperforming them with respect to AP_ϵ (see Table 5.5), indicating that it is more sensitive in localizing PL while maintaining a high level of precision at the same time.

After characterizing localization errors, the next step is to investigate the impact of localization errors on the performance of PL localizers. Correcting localization errors would result in an improvement of AP_ϵ since more TP predictions (hence fewer FP predictions) are identified. Figure 5.12 shows the improvement of AP_ϵ ($\epsilon \in \{0.3, 0.5, 0.7\}$) obtained by correcting *jitter* errors and *miss* errors respectively. Note that there is no improvement for $AP_{0.3}$ in Figure 5.12a since all *jitter* errors are identified as TP predictions for $\epsilon = 0.3$ even without correction. As expected, the impact of *jitter* errors is related to SOKS levels as shown in Figure 5.12a. The reason is that there are more corrected *jitter* errors at a higher level than at a lower level according to Equation 5.13, hence more AP improvement would be achieved

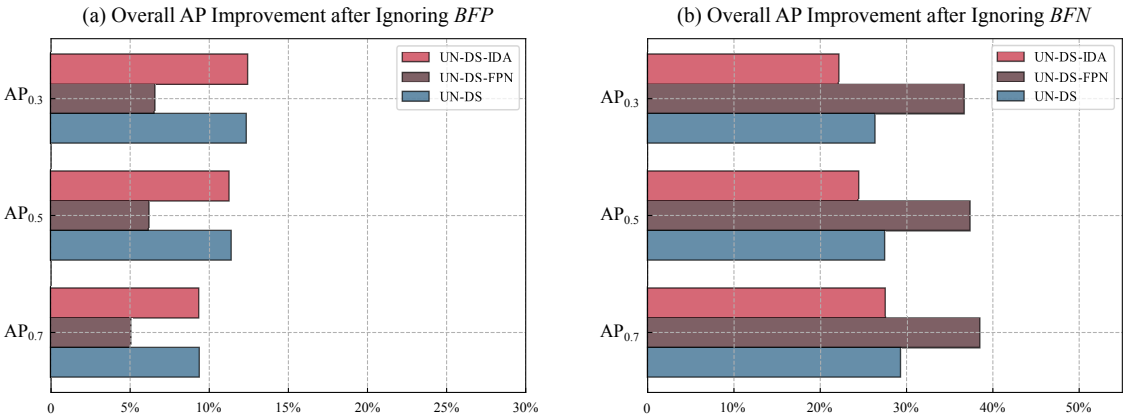


Figure 5.15: Overall AP improvements obtained after ignoring background errors; evaluated at SOKS levels of {0.3, 0.5, 0.7}.

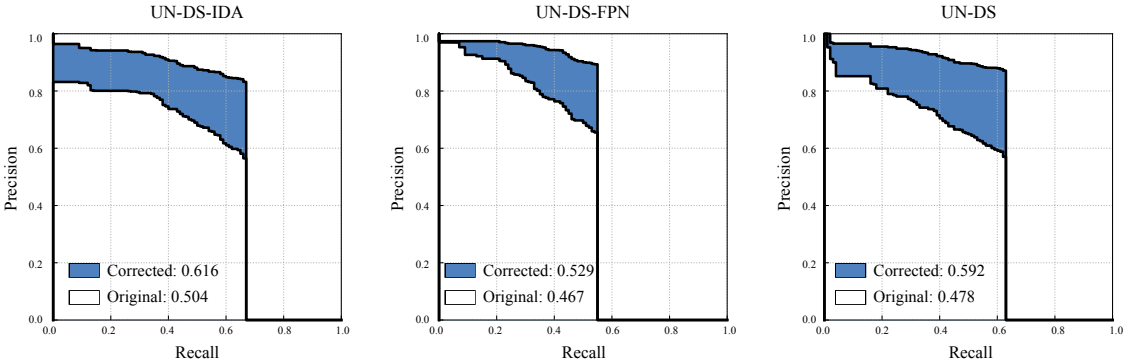


Figure 5.16: Precision-recall curve changed after ignoring BFP errors; evaluated at SOKS level of 0.5.

at a higher SOKS level. Conversely, the impact of *miss* errors remains stable at different SOKS levels (Figure 5.12b). This is because the number of corrected *miss* errors is the same for $\epsilon \in \{0.3 : 0.05 : 0.95\}$ according to Equation 5.14. Take the improvement of $AP_{0.5}$ as an example. Figure 5.13 and Figure 5.14 present the change of precision-recall curves before and after correcting *jitter* and *miss* errors respectively. In general, *jitter* errors contain many PL predictions with high confidence scores, which is revealed by the improvement along the curve in the top-left part of the precision-recall plots after error correction. By contrast, *miss* errors contain mostly PL predictions with low confidence scores, as illustrated by the small improvement along the curve in the top-left part of the plots.

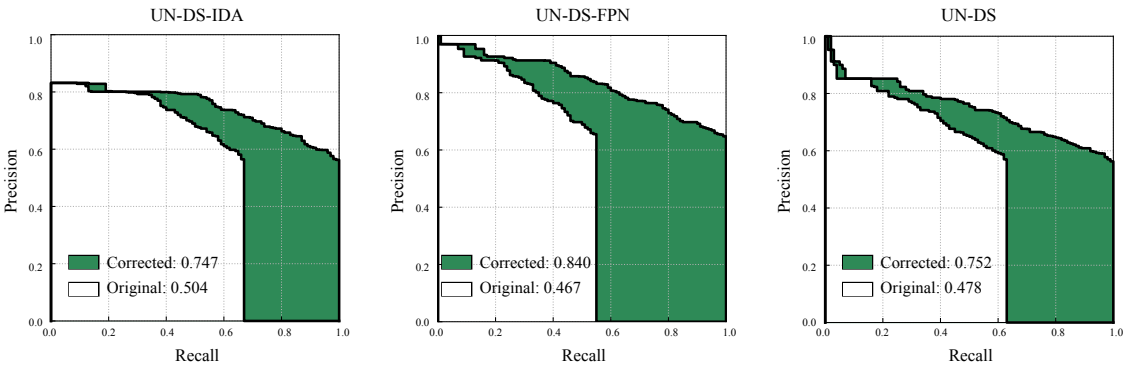


Figure 5.17: Precision-recall curve changed after correcting BFN errors; evaluated at SOKS level of 0.5.

(ii) Background Errors

A background error occurs when the matching process in § 5.4 fails to pair a PL prediction with any reference or conversely. Two types of background errors are defined, as shown in Figure 5.10b.

- *Background false positive (BFP)*: an unmatched PL prediction after AP evaluation is performed.
- *Background false negative (BFN)*: an unmatched reference PL after AP evaluation is performed.

BFP errors typically occur in two circumstances: (1) a poor PL prediction with a SOKS score smaller than 0.05 to its nearest reference, such as the false positive PL predictions in Figure 5.1; (2) a PL prediction that is not matched because its nearest reference PL has been matched to a prediction with a higher SOKS score. To deal with the first circumstance, a better localization model that captures global contextual cues is needed to filter out false positive PL predictions that do not follow the placental geometry, e.g. getting rid of predictions that lie outside the placenta itself. The second circumstance may be resolved by performing better non-maximum-suppression (NMS). On the other hand, *BFN* errors are mainly caused by the difficulty in identifying PLs of small sizes and low contrast.

As with localization errors, the impact of background errors is examined by ignoring *BFP* and *BFN* errors respectively at AP evaluations, as shown in Figure 5.15.

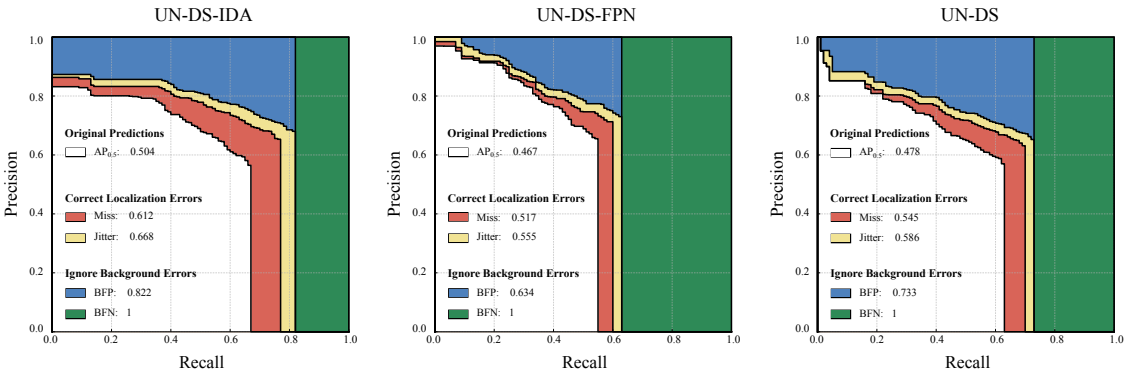


Figure 5.18: Coarse to fine error analysis. The numbers in each plot are the value of $AP_{0.5}$ after sequentially fixing each error type in the following order: *miss, jitter, BFP, BFN*.

It is demonstrated that *BFN* errors affect localization performance significantly more than *BFP* errors, which implies that there is still a large room for improvement of the examined PL localizers. Figure 5.16 and Figure 5.17 display the change of precision-recall curves before and after correcting *BFP* and *BFN* errors respectively. UN-DS-IDA made fewest *BFN* errors compared to the other two localizers, which contributes to its best localization performance overall. The precision-recall curves evaluated at SOKS level of 0.5, after sequentially correcting localization errors and background errors, are displayed in Figure 5.18.

5.6.2 Sensitivity to Instance Number

Previous analysis on localization errors and background errors reflects the performance of the examined PL localizers over the entire dataset. This subsection intends to understand how the characteristics of placental ultrasound images such as instance number influence localization performance, which can further help understand PL localizers' behaviours and identify their strengths and weaknesses.

An instance number IN_i is an image-level feature that describes the number of identified PL instances in a placenta ultrasound image i. It has been observed that numerous PLs are likely to be found in PASD cases because of the abnormal vasculature development of the placenta in the presence of this disorder (see Section 1.2.2). Consequently, it is clinically relevant to investigate the performance

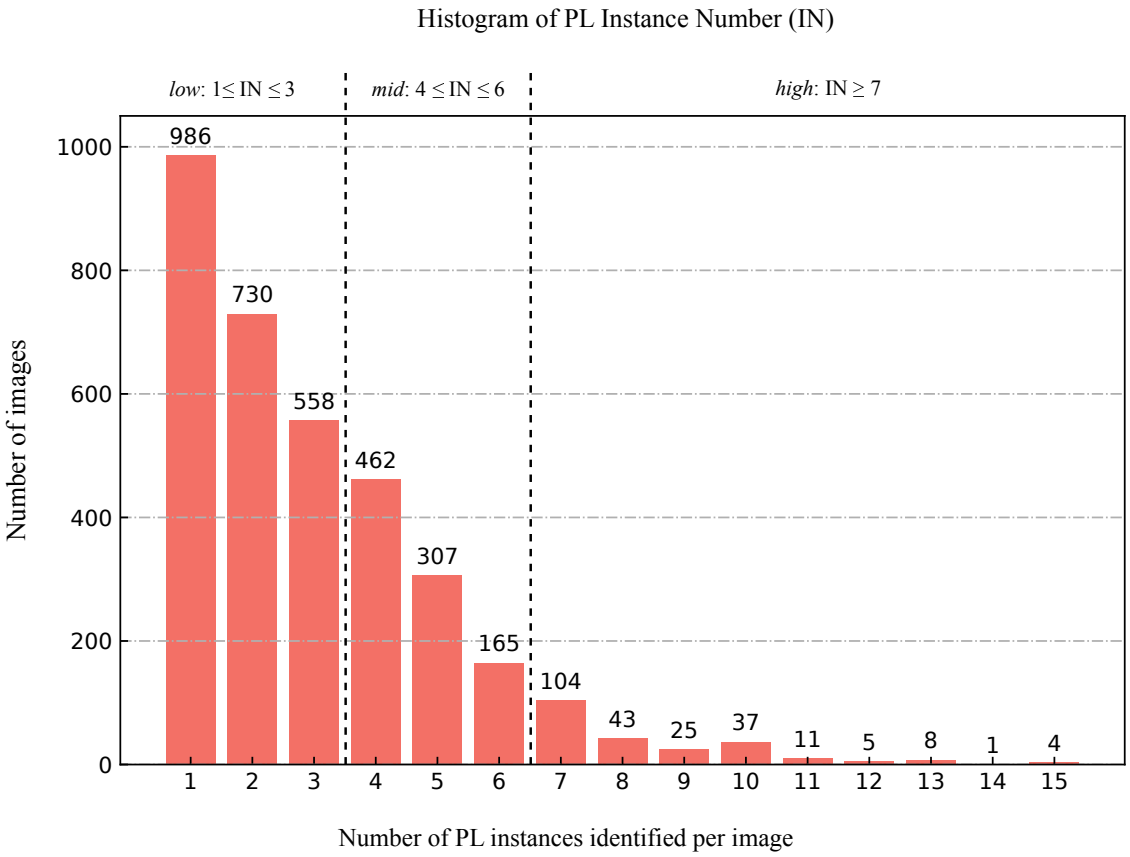


Figure 5.19: Distribution of the number of PL instances per image over the entire dataset.

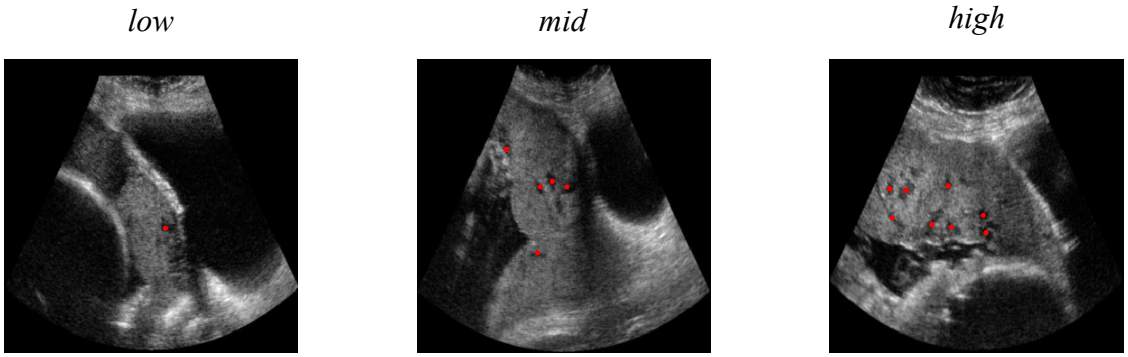


Figure 5.20: Sample images with dot annotations from three subsets.

of a PL localizer when dealing with images of different instance numbers, especially those with a high instance number.

Figure 5.19 presents the histogram of PL instance numbers over the dataset. It shows that around 66% of the images contain $1 \sim 3$ annotated PL instances, 27.1% contain $4 \sim 6$ PL instances and only 6.9% contain more than 6 PL instances.

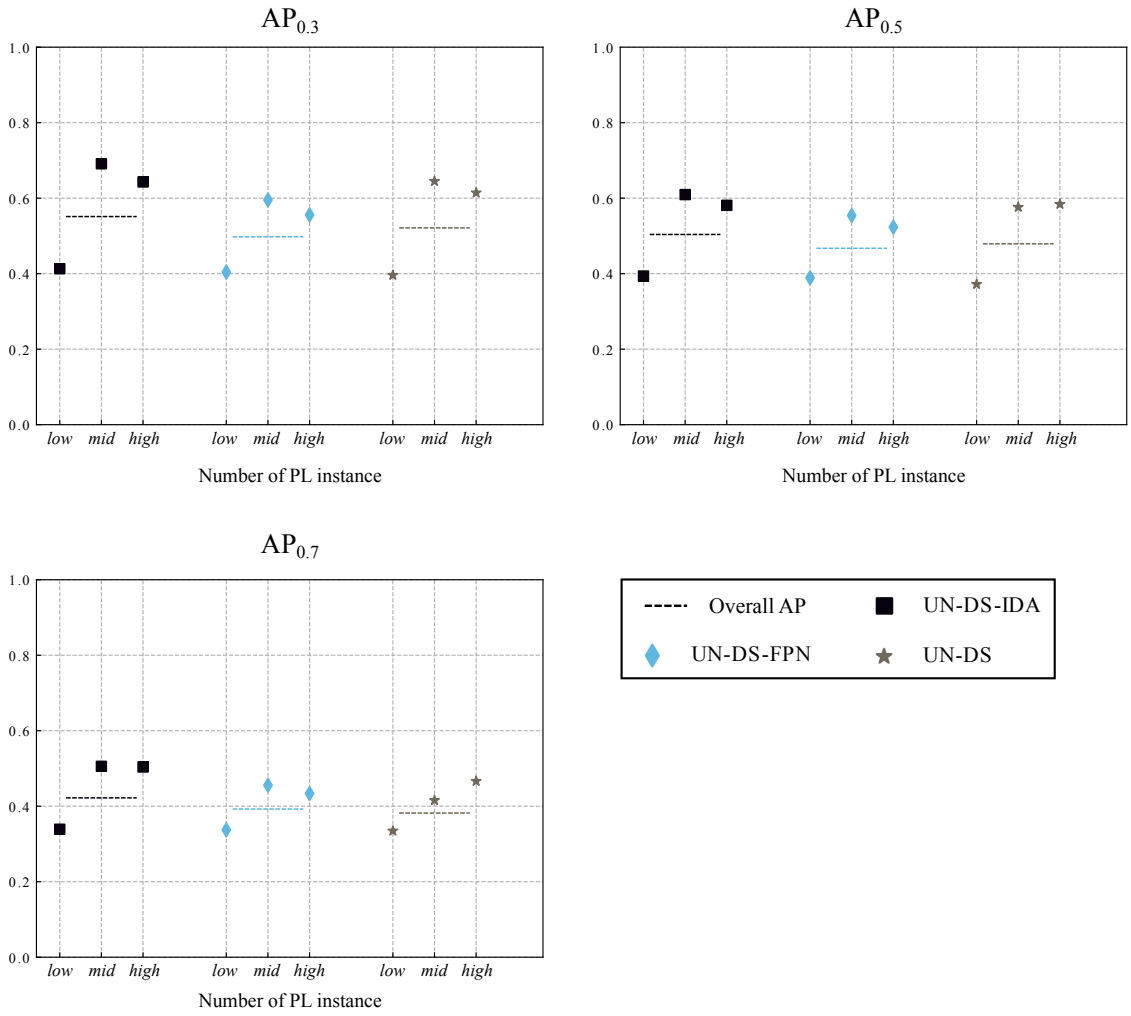


Figure 5.21: Model sensitivity to instance number, evaluated at SOKS levels of {0.3, 0.5, 0.7}.

Consequently, the dataset is further divided into three non-overlapping subsets based on the instance number of each image. Sample images from these subsets are shown in Figure 5.20. The subset named *low* contains images with instance number $1 \leq \mathrm{IN}_i \leq 3$; the subset named *mid* contains images with $4 \leq \mathrm{IN}_i \leq 6$; the subset named *high* contains images with $\mathrm{IN}_i \geq 7$. Based on this division, AP evaluation and error analysis are performed to test three PL localizers' sensitivity to instance number. As with previous analysis, three PL localizers are trained on fold-1. The test set of fold-1 has a *low* subset with 210 images, a *mid* subset with 64 images, and a *high* subset with 25 images.

Figure 5.21 displays evaluation results on fold-1, with three SOKS levels

Table 5.6: Frequency of Localization Errors Occuring on Three Subsets.

Error Type	UN-DS-IDA			UN-DS-FPN			UN-DS		
	low	*mid*	*high*	*low*	*mid*	*high*	*low*	*mid*	*high*
Acceptable	173	148	69	139	117	62	171	130	69
Miss	34	115	80	15	91	59	10	110	63
Jitter	29	44	12	22	32	7	26	44	11

$\epsilon \in \{0.3, 0.5, 0.7\}$. It demonstrates that all three PL localizers have the worst performance on *low* subset, with AP≈ 0.4 on average. One explanation is that AP evaluation is more vulnerable to false positive predictions when the instance number is low. Another reason lies in the fact that PL instances with smaller scales are more prevalent in *low* subset than in *mid* and *high* subsets because of the vascular effect. Since it is already hard to achieve a high SOKS score for smaller-scale PL instances compared to larger-scale instances as defined in Equation 5.6, localization is more challenging in *low* subset when evaluated on AP. As a result, AP evaluation is more satisfying on *mid* and *high* subsets for all three models.

It is worth noting that UN-DS-IDA outperforms the other two PL localizers on both *mid* and *high* subsets, which are more clinically relevant than *low* subsets. On one hand, UN-DS-IDA made more *acceptable* predictions (see Table 5.6) than the rest two models, indicating that it produces more true positive predictions. On the other hand, the total number of *acceptable, miss, jitter* errors of UN-DS-IDA is also significantly higher, which indicates its superior performance in terms of sensitivity (recall), i.e. finding more relevant PL instances.

5.6.3 Limitation

As stated in Section 3.2.3, performance evaluation using dot annotations are inclined to subjectivity as positions of PL *centroids* determined by visual inspection can vary across annotators. SOKS scores are proposed to alleviate influence of inter-rate variability by controlling the falloff in Equation 5.6. Nevertheless, the absence of annotation from other expert annotators, who can produce slightly diverse but plausible dot annotations, remains a limitation of this chapter. If available, statistical analysis can be performed to determine reliability of the proposed

PL localization models compared to human experts. Moreover, recent work on probabilistic modelling of multiple annotations may further improve localization performance by producing more plausible results when facing reference ambiguity during training [144]. These shall be a direction for future work.

5.7 Conclusions

This chapter has presented a deep layer aggregation approach to PL localization, which effectively fuses features from multiple network scales via non-linear pathways. The supervision signal associated with each placental ultrasound image is a confidence map, which was generated from experts' dot annotations on PL centroids. It was found that the proposed map generation method can adaptively capture the shape and size information of a PL. Two types of network architectures were tested: the downsample network (DN) and the U-shape network (UN). The proposed deeply supervised iterative aggregation structure (DN-DS-IDA & UN-DS-IDA) and deeply supervised feature pyramid network (DS-DS-FPN & UN-DS-FPN) were trained to regress PL confidence maps via a combination of a fusion loss and side-output losses. Inspired by the evaluation framework of human pose estimation, a metric named simplified object keypoint similarity (SOKS) was proposed to measure how good a PL prediction was with respect to a reference location. Based on SOKS, AP evaluation was used to evaluate model performance in terms of PL localization. It was found that UN-DS-IDA achieved the best result by effectively aggregating features from shallower stages, which contain important low-level features related to PL.

To further investigate the cause of errors, a detailed error analysis framework has been presented. In particular, the framework defined two types of errors for PL localization: localization errors and background errors. The former characterized predictions that were successfully matched to PL references with different SOKS scores, which can be further categorized into *jitter*, *miss* and *acceptable*. The latter dealt with unmatched predictions (background false positive, or BFP) and unmatched references (background false negative, or BFN). It was found that *miss* errors are mostly PL predictions with low confidence scores while *jitter* errors are

mostly those with high confidence scores. Moreover, BFN errors were predominant in all examined PL localizers, which were mainly caused by the difficulty in identifying PLs of small sizes and low contrast. Sensitivity analysis on instance number revealed that the proposed models were better at localizing images with higher instance numbers, which were more clinically relevant because of the vascular effect of PASD. It was found that UN-DS-IDA was generally better at finding more relevant PL instances than the rest models.

6

Utero-placental Interface Detection in Placental Ultrasound

Contents

6.1 Background and Motivation 97
6.2 UPI-Net: Semantic Edge Detection in Placental Ultrasound 102
6.2.1 Problem Formulation 102
6.2.2 Network Architecture 104
6.3 Experiments and Analysis 109
6.3.1 Dataset 109
6.3.2 Evaluation Protocol 109
6.3.3 Evaluation Metrics 110
6.3.4 Hyperparameter Tuning 111
6.3.5 Implementation Details 112
6.3.6 Results 116
6.3.7 Limitation 117
6.4 Conclusions 118

This chapter presents machine learning algorithms that address the task of utero-placental interface (UPI) detection in placental ultrasound images, whereby it becomes tangible to extract and interpret PASD-related sonographic patterns near the UPI (see § 1.2.2). Deep convolutional neural networks (CNN) have demonstrated potential for general detection and segmentation tasks in cluttered environments. In effect, UPI detection is formulated as a semantic edge detection problem that is

approached by predicting each image pixel's probability of locating on the UPI. This is a particularly challenging task owing to the indistinct contrast of the placenta boundary and strong variability in the shape and size of UPI. Moreover, as opposed to classic (semantic) edge detection in computer vision, placental ultrasound images contain specific anatomical structures arranged in a unique geometry. This chapter argues that it would be beneficial for UPI detectors to exploit global contextual cues of placenta ultrasound images to filter out unwanted false positive predictions. Whilst this chapter deals with placental ultrasound, it is believed that the proposed approach could be adapted for other clinical scenarios that involve edge/contour detection such as analysis of breast, liver, heart and brain imaging.

The majority of work presented in this chapter were published and presented as an oral report at the Visual Recognition for Medical Images Workshop, 2019 International Conference on Computer Vision (ICCV), Seoul, South Korea [145]. This chapter extends the work in that paper by incorporating more experimental results that highlight characteristics of various UPI detectors in terms of design choices, training losses and ablation studies.

6.1 Background and Motivation

Edge/contour detection has been a fundamental task in the field of computer vision since the early 1970's [146, 147]. The detection of edges acts as a key first step for a variety of vision tasks, such as image segmentation [148, 149] and object detection [150, 151]. Traditional approaches exploit low-level gradients and non-maximal suppression techniques [152–154] to locate edge proposals. Combinations of more complex visual features including brightness, color, texture and depth gradients can also be incorporated for robust edge detection [148, 155, 156]. The use of machine learning techniques have also been explored for edge detection, such as probabilistic boosting trees [157], random decision forests [158] and deep convolutional neural networks [140, 159–164]. In recent years, a series of CNN-based algorithms have achieved state-of-the-art performance in edge detection benchmark datasets such as the Berkeley Segmentation Data Set and Benchmarks 500 (BSDS500) [148],

NYU Depth dataset [165], multi-cue boundary detection dataset [166]. Instead of manually constructing edge-sensitive features, CNN-based models utilize a stack of non-linear hidden layers to learn complex feature representations directly from raw data. This chapter first reviews CNN-based models that deal with two types of edge detection tasks: (i) class-agnostic edge detection; and (ii) semantic edge detection.

(i) Class-agnostic edge detection. This corresponds to the traditional edge detection task, where perceptual edges of objects instances are the main focus. It does not discriminate between boundaries of different objects, hence the term *class-agnostic*. Bertasius et al. proposed a DeepEdge model that exploited object-related high-level cues to boost edge detection, a task that is often approached with low-level cues. Without any feature engineering, their multi-scale bifurcated network outperformed other learning-based or non-learning methods [167]. Xie et al. proposed a fully convolutional edge detection model named holistically-nested edge detector (HED) that was trained end-to-end to predict the pixel-wise edge likelihood powered by deep supervision [168]. Training of HED was regularized by forcing intermediate layers to produce edge predictions (side outputs) [140]. Wang et al. proposed a top-down backward refinement pathway inside an FCN that would promote *crisp* edge predictions with less blurry appearance and better localization performance [163]. He et al. further enriched multi-scale representations via dilated convolutions in a bi-directional cascade network [161]. In general, recent work has been investigating architectural improvements for better multi-scale representations.

(ii) Semantic edge detection. Assigning semantics to edges is a relatively new task that has not received much attention. Early work used class-specific edges for tracking [157, 169], object detection and segmentation [170]. Hariharan et al. presented the large-scale Semantic Boundaries Dataset (SBD) and proposed to use generic object detectors along with bottom-up contours for semantic contour detection [171]. See Figure 6.2 for samples from SBD. Bertasius et

al. introduced a CNN-based two-stage process that first identified all edge candidates and then classified them using segmentation networks [79, 172, 173]. Yu et al. proposed CASENet to detect semantic edges in an end-to-end fashion. They optimized HED [140] by removing deep supervisions on the early-stage side outputs and instead used them as shared features for the final fusion [174]. Liu et al. proposed a diverse deep supervision mechanism, whereby lower layers aimed at generating class-agnostic edges and higher layers targeted class-aware semantic edges [175]. Acuna et al. regularized edge predictions using a differentiable non-maximum-suppression loss to produce thinner edges [176]. Overall, approaches to these two types of edge detection tasks have a lot in common in terms of problem formulation, choice of objective functions and network architectures.

In order to recognize edge pixels of specific semantic categories, CNNs are often designed to have large receptive fields by repeatedly stacking downsampling and (dilated) convolution layers [84, 177–179], which is reported to be computationally inefficient and difficult to optimize in general [180, 181]. To address this issue, a self-attention mechanism, originally born in natural language processing studies [182], can be introduced to explicitly model element-wise correlation [180, 183] and has achieved success in various visual recognition applications such as video classification [180], object detection [184], semantic segmentation [185], panoptic segmentation [186], generative adversarial networks [183], and representation learning [181, 187–191]. It is recently reported that non-local pixel-wise attention can be simplified as a more memory-efficient query-independent attention without sacrificing performance [180, 181]. Consequently, incorporating global context modelling into an existing semantic edge detector is expected to improve its discriminative capability for high-level semantics in a more effective manner.

UPI detection in placental ultrasound images falls in the category of semantic edge detection. A good detector should distinguish the UPI from false positive edge segments that only resemble the appearance of UPI (see Figure 6.1 for examples). In normal cases, the UPI is observed as the placental boundary that touches

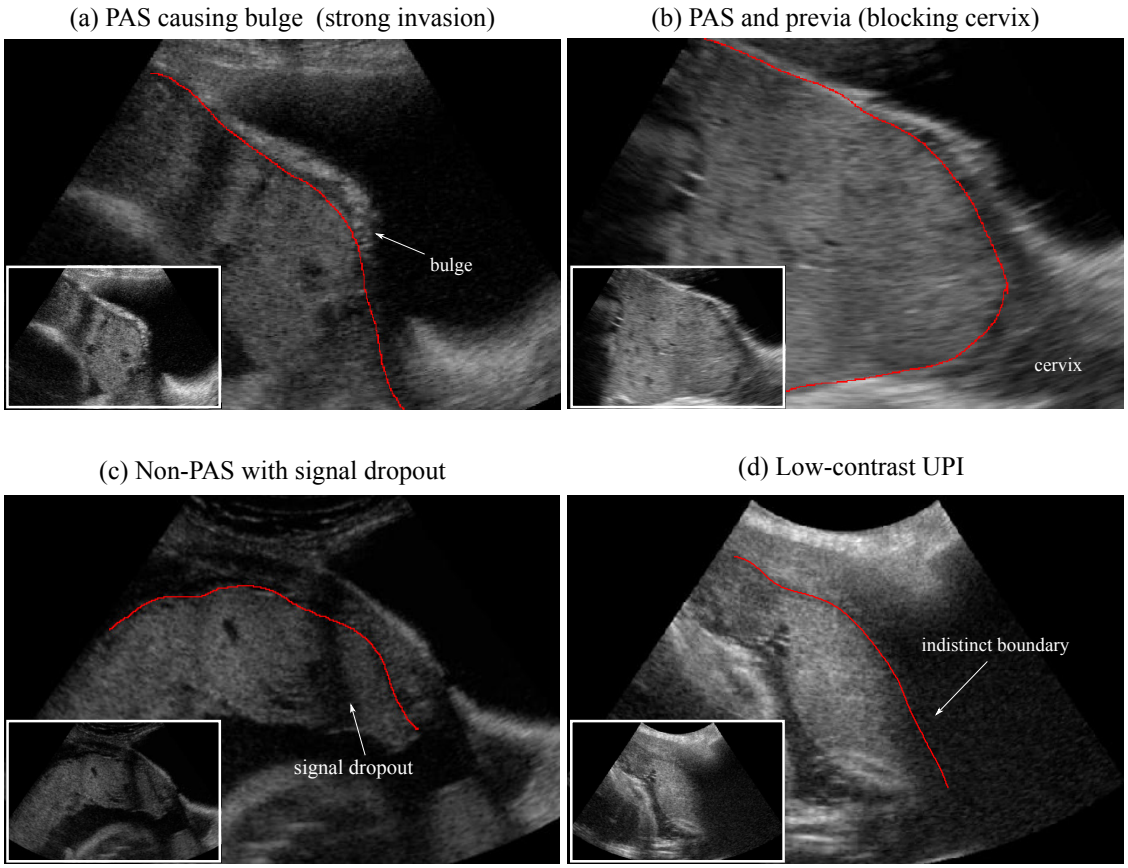


Figure 6.1: Utero-placental interfaces (UPI) are annotated as red curves in placental ultrasound image examples. In PASD cases, the myometrium tends to disappear due to placental invasion, causing weaker contrast around the UPI. Strong placental invasion would produce a bulge-like UPI as shown in (a). In the placenta previa case, UPI usually takes a 'U-shape' over the cervix, as shown in (b). A non-PAS case where UPI separates the placenta from the myometrium is shown in (c), somewhat contaminated by signal dropout. An example with a low-contrast UPI is shown in (d). The indistinct appearance makes detection more challenging.

the myometrium. However, in PASD cases, the degree of placental invasion can vary along the UPI, resulting in an irregular shape and length with low contrast. Manual localization remains challenging and time-consuming even for experienced sonographers, as shown in Figure 6.1.

Figure 6.2 displays example images from SBD [171] and the Oxford-Hackensack database of PASD (see Chapter 3) respectively. In natural images, objects of interest may appear with various scales at different locations within a scene. More often than not, the network receptive field is large enough to capture relevant semantics for semantic contour detection. On the contrary, placental ultrasound

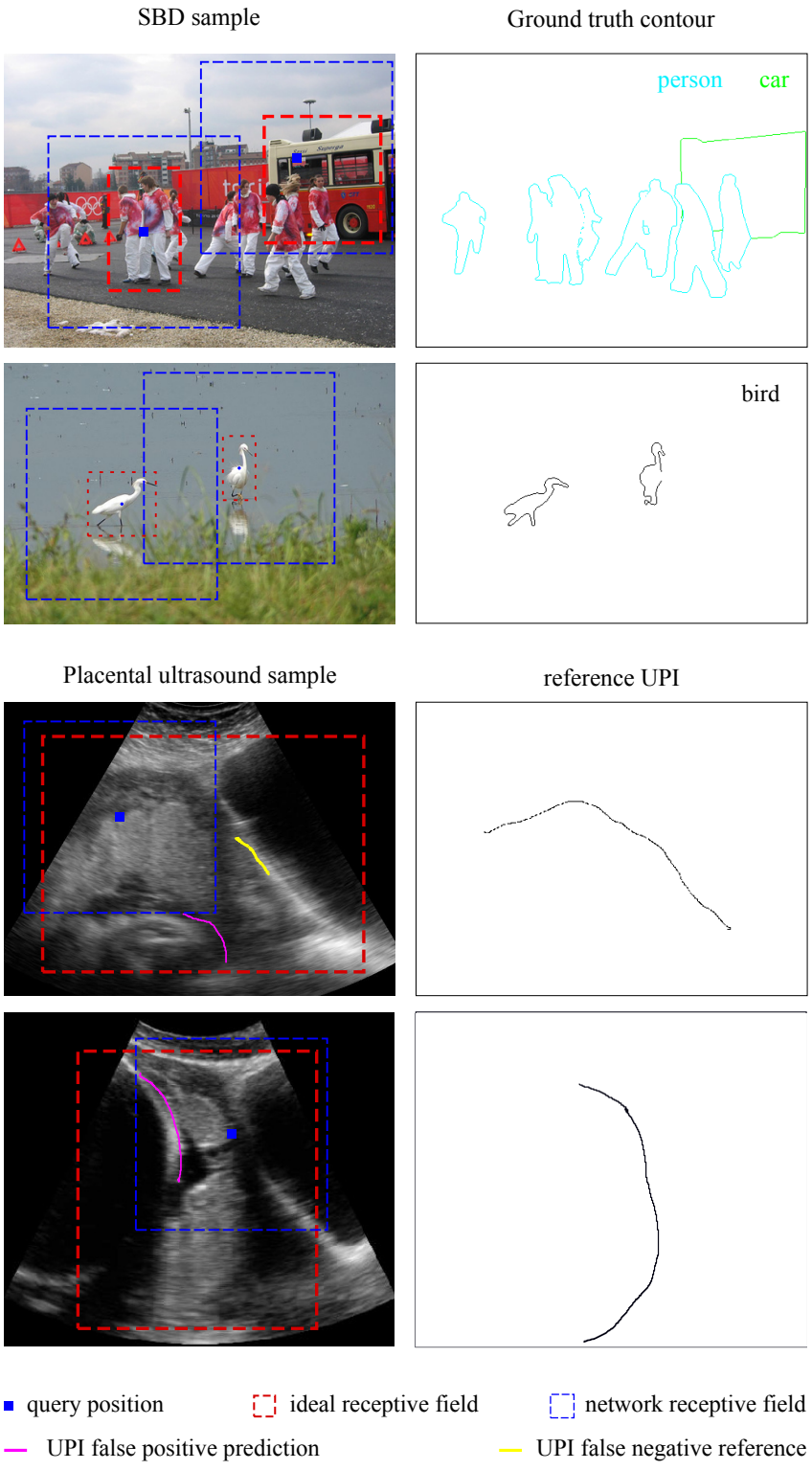


Figure 6.2: Semantic edge detection in natural images (examples from SBD [171]) and in placental ultrasound images (examples from the Oxford-Hackensack database of PASD (see Chapter 3). Best viewed in color.

images contain specific anatomical structures thus have unique geometry. From a low-level perspective, there are a considerable number of UPI-like edges (false positives, e.g. in Figure 6.2). It is necessary to suppress these irrelevant edges that are not parts of UPI (i.e. do not separate the placenta from the uterus) by modelling high-level semantics, which requires the network to also identify specific semantic entities related to placenta geometry [188, 192]. This chapter introduces methods for modelling global context of each spatial position in order to suppress false predictions thus improve UPI detection performance. The resulting model, named UPI-Net, is a deep network designed for UPI detection in placental ultrasound images. UPI-Net captures the long-range dependencies in placenta geometry using lightweight global context modelling units and effective multi-scale feature aggregation. The contributions are twofold. First, a novel architecture is presented to enforce contextual feature learning in the earlier stages and to enhance learning of UPI-related semantic entities / geometry in later stages. Second, the performance of UPI-Net is evaluated using standard edge/contour detection metrics [148, 171] and compared against several competitive benchmarks on a placental ultrasound image collection. According to the experiments, UPI-Net yields the best performance without introducing considerable computational overhead.

6.2 UPI-Net: Semantic Edge Detection in Placental Ultrasound

This section starts with the problem formulation, followed by the introduction of two convolutional blocks that facilitate global context modelling and high-level representation learning. The complete architecture of UPI-Net is presented at the end.

6.2.1 Problem Formulation

Training process. Let the training set be denoted as $D = \{(\mathcal{X}_n, \mathcal{Y}_n), n = 1, \cdots, |D|\}$, where a sample $\mathcal{X}_n = \{x_n^p, p = 1, \cdots, |\mathcal{X}_n|\}$ is a placental ultrasound image and $\mathcal{Y}_n = \{y_n^p, p = 1, \cdots, |\mathcal{X}_n|\}$ denotes the corresponding reference UPI map for $\mathcal{X}_n$. In practice, $\mathcal{Y}_n$ can take a variety of forms, such as a confidence map or a

binary mask. Take a binary mask as an example, with $y_n^p \in \{0, 1\}$, i.e. pixels on the UPI take the value 1 and the rest take 0. For notation simplicity, the subscript n is dropped from now on. The overall goal is to train a network Ψ with parameters $\mathbf{W}$ to predict the probability map $\hat{\mathcal{Y}} = \{\hat{y}^p, p = 1, \cdots, |\mathcal{X}|\}$, where $\hat{y}^p$ is the network estimation of y^p at each pixel position p. Following [140, 174], a class-balancing weight ω is introduced to alleviate the extremely low foreground-background class ratio encountered during training. This is based on the idea of prior scaling [193], with the purpose to equalize the expected model weight update for both classes. Specifically, the following cross-entropy loss function is defined for the network output $\mathcal{O} = \Psi(\mathcal{X})$ given a training pair $(\mathcal{X}, \mathcal{Y})$:

$$L(\mathcal{O}; \mathbf{W}) = -\frac{\omega}{|\mathcal{Y}|} \sum_{p \in |\mathcal{Y}_+|} \log(\hat{y}^p) - \frac{1}{|\mathcal{Y}|} \sum_{p \in |\mathcal{Y}_-|} \log(1 - \hat{y}^p) \tag{6.1}$$

where ω is set to $\frac{|\mathcal{Y}_-|}{|\mathcal{Y}_+|}$, with $|\mathcal{Y}_+|$ and $|\mathcal{Y}_-|$ being the number of positives (i.e. pixels on the UPI) and negatives (i.e. pixels off the UPI). $|\mathcal{Y}|$ is the total number of pixels within an image, hence $|\mathcal{Y}| = |\mathcal{Y}_+| + |\mathcal{Y}_-|$. The network output $\mathcal{O}$ at pixel position p is activated by a sigmoid function to obtain $\hat{y}^p$:

$$\hat{y}^p = \frac{1}{1 + \exp(-\mathcal{O}^p)} \tag{6.2}$$

UPI-Net has two outputs, a side output $\mathcal{O}_s$ and a fused output $\mathcal{O}_f$. The details will be discussed in § 6.2.2. Each output corresponds to an individual prediction. The overall loss function is simply the sum of losses on individual outputs:

$$L_{all}(\mathbf{W}) = L(\mathcal{O}_s; \mathbf{W}) + L(\mathcal{O}_f; \mathbf{W}) \tag{6.3}$$

Testing process. During testing, UPI-Net would output two predictions given an input placental ultrasound image $\mathcal{X}$. The final prediction is simply the sigmoid of the fused output,

$$Out = \frac{1}{1 + \exp(-\mathcal{O}_f)} \tag{6.4}$$

All the evaluations and hyper-parameter searches are based on this prediction alone, whilst optimizing the side output $\mathcal{O}_s$ contributes to network regularization [168].

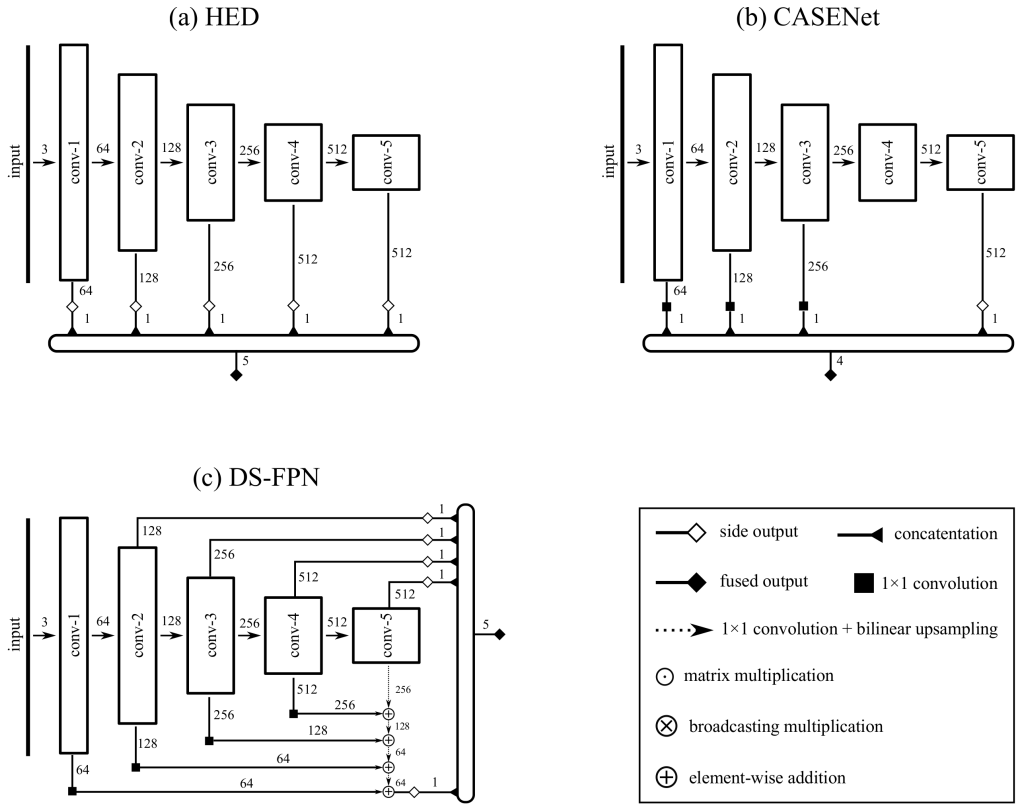


Figure 6.3: Three multi-scale feature aggregation architectures: (a) HED [140]; (b) CASENet [174]; (c) DS-FPN [116].

6.2.2 Network Architecture

Rich hierarchical representations of deep neural networks lead to success in edge detection [140, 174]. This is particularly important for UPI detection, which requires effective aggregation of multi-scale features to localize edge pixels on the UPI and get rid of false positives using global context of placenta geometry. This subsection first presents three multi-scale feature aggregation architectures that have been proposed in the literature for edge detection and key-point localization [116, 140, 174, 194]. Their suitability for UPI detection is discussed.

Multi-scale Feature Aggregation

Three popular deep network architectures that aggregate multi-scale features are shown in Figure 6.3: HED [140], CASENet [174], and DS-FPN [116] (see also Chapter 5). To be structurally consistent, they are all built upon the VGG-16 network [83]. HED inherits the idea of deeply-supervised nets [168] to produce

five individual side outputs at different scales and one fused output by merging five side outputs via concatenation and 1×1 convolutions. CASENet adopts a similar nested architecture but disables early-stage deep supervisions thus only produces one side output and one fused output. DS-FPN extends the idea of feature pyramid networks [194] by merging multi-scale features via 1×1 convolutions and element-wise additions, producing five side outputs and one fused output.

UPI detection depends both on low-level features associated with edges, which are well preserved in the shallower stages of the network, and on high-level semantic entities associated with placenta geometry, which are learnt in the deeper stages of the networks. One common issue related to the three architectures above is the sub-optimal use of low-level features. Previous work tends to use them for feature augmentation without careful refinement. This chapter argues that it is beneficial for UPI detectors to incorporate global context modelling in features of different scales (esp. those in the shallower stages). After all, large receptive fields are only available in the deepest stages of the networks via stacked convolutional operations, which might not even be large enough to model important long-distance dependencies in placental ultrasound images, as discussed in § 6.1.

UPI-Net: Detecting the UPI in Placental Ultrasound Images

The proposed UPI-Net (Figure 6.4) aims to address the aforementioned issues by adding two types of feature refinement blocks in a fully convolutional neural network: (i) global context (GC) blocks [181]; and (ii) convolutional group-wise enhancement (CGE) blocks.

- **GC blocks**. A GC block modulates low-level features via simplified non-local operations and channel recalibration operations. Let $\mathcal{F} \in \mathbb{R}^{B \times C \times H \times W}$ denote an intermediate feature map of a CNN. It is represented as a four-dimensional tensor with a mini-batch size of B, a channel number of C and a spatial resolution of $H \times W$. As shown in Figure 6.5, a GC block first performs global attention pooling on the input feature map $\mathcal{F}$ via a 1×1 convolution and a spatial softmax layer. The output is then multiplied with the original

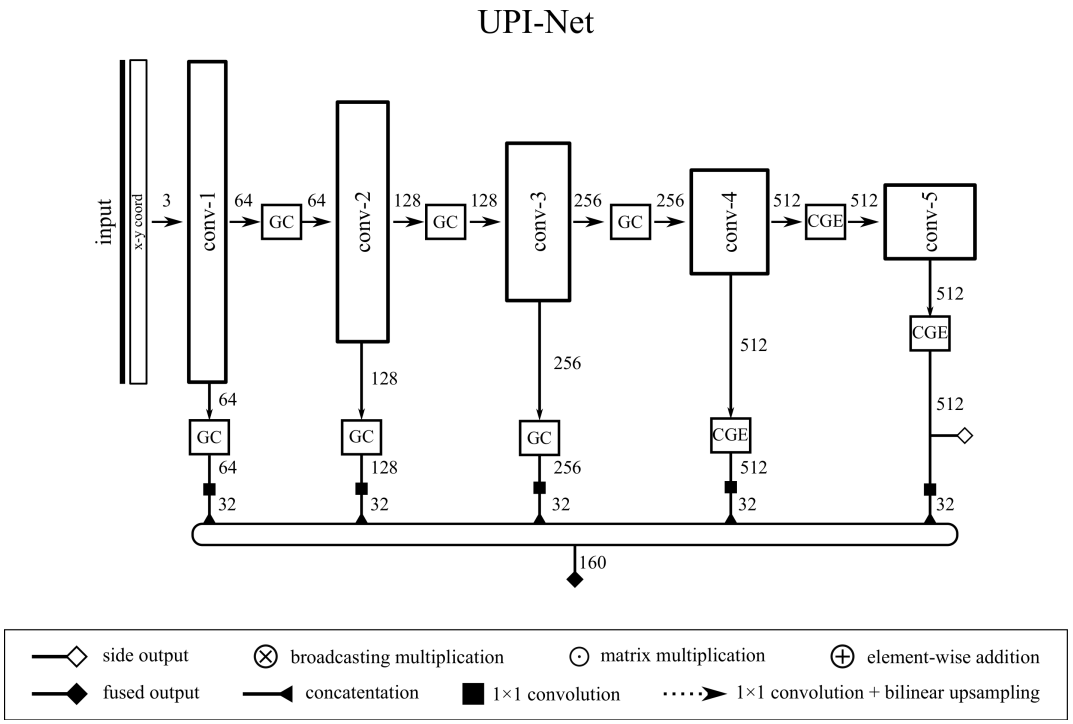


Figure 6.4: The proposed UPI-Net with a VGG-16 backbone.

input to obtain a channel attention weight of size $B \times C \times 1 \times 1$. After a channel recalibration transform (via a squeeze-and-excitation block with a squeezing ratio $r = 16$ [187]), the calibrated weight is aggregated back to the original input via a broadcasting addition to obtain a refined feature map $\mathcal{F}' \in \mathbb{R}^{B \times C \times H \times W}$ of the same size as the input. As reported in [181], a GC block is a lightweight alternative to the non-local block [180] in modelling global context of the input feature map. In UPI-Net, GC blocks are attached to *conv-1*, *conv-2* and *conv-3* to refine features from the earlier stages of the network through global modelling.

- **CGE blocks**. Inspired by [188], this section proposes a convolutional group-wise enhancement (CGE) block to promote learning of high-level semantic entities related to UPI detection via group-wise operations. As shown in Figure 6.6, a CGE block contains a group-convolution layer with the group number N_G, a group-norm layer [70] also with the group number N_G, and a sigmoid function. The group convolution layer essentially splits an input

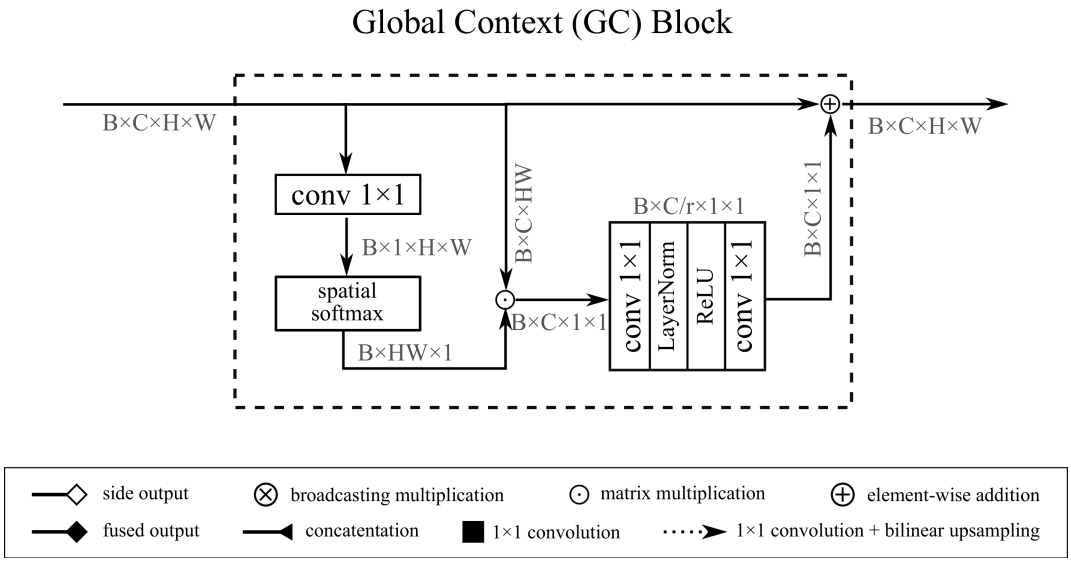


Figure 6.5: A global context (GC) block, which takes a $B \times C \times H \times W$ feature map and produces a refined feature map of the same size.

feature map $\mathcal{F} \in \mathbb{R}^{B\times C\times H\times W}$ into N_G groups along the channel dimension. After convolution, each group contains a feature map of size $B\times 1\times H\times W$. The subsequent group-norm layer normalizes each map over the space respectively. The learnable scale and shift parameters in group-norm layers are initialized to ones and zeros following [70]. The sigmoid function serves as a gating mechanism to produce a group of *importance maps*, which are used to scale the original inputs via the broadcasting multiplication. It is expected that the group-wise operations in CGE would facilitate the formation of unique semantic entities across groups. The group-norm layer and sigmoid function can help enhance UPI-related semantics by suppressing irrelevant noise. The proposed CGE block is a modified version of the spatial group-wise enhance (SGE) block in [188] by replacing the original attention pooling with a simple 1×1 group convolution as learnable weights tend to be more expressive than weights from global average pooling in capturing high-level semantics. Experiments on the validation set empirically support this design choice. CGE blocks are attached to *conv-4* and *conv-5* respectively, where high-level semantics are learnt.

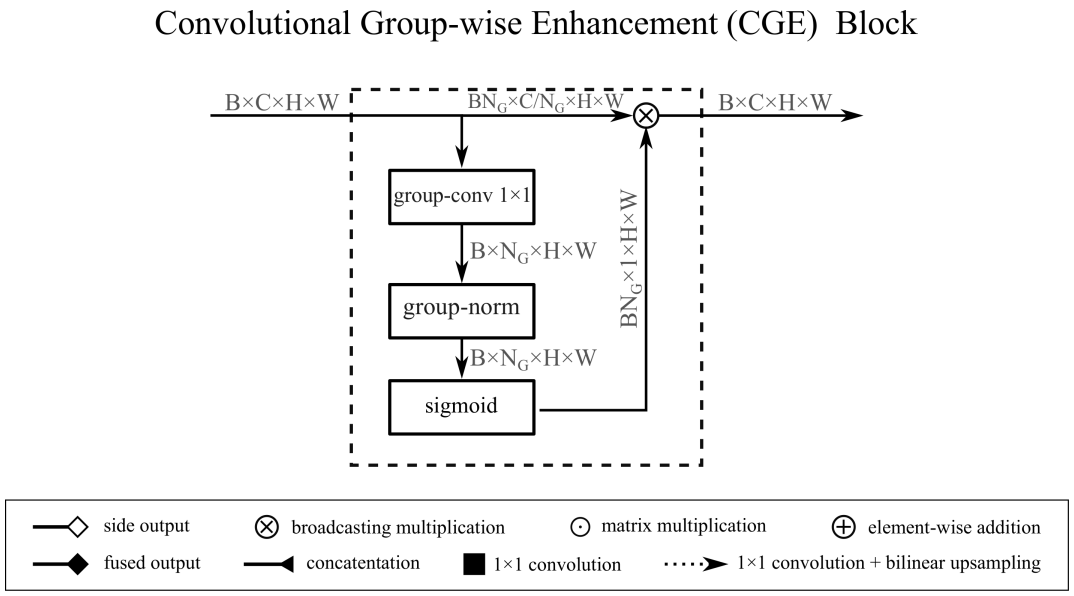


Figure 6.6: A convolutional group-wise enhancement (CGE) block, which takes a $B \times C \times H \times W$ feature map and produces a refined feature map of the same size.

As displayed in Figure 6.4, all refined features are linearly transformed via 1×1 convolution to have $N_C = 32$ channels. They are then aggregated via channel-wise concatenation to produce the fused output. Additionally, a side output is produced from *conv-5*, which encodes strong high-level semantics. All the channel mismatches are resolved by 1×1 convolution and resolution mismatches by bilinear upsampling. A Coord-Conv layer [195] is added in the beginning of the UPI-Net, which simply requires concatenation of two layers of coordinates in (x, y) Cartesian space respectively. Coordinates are re-scaled to fall in the range $[-1, 1]$. It is expected that the Coord-Conv layer would promote the implicit learning of placenta geometry by enriching spatial representations. Another desirable property of the Coord-Conv layer is that it does not add computational cost to the network[1]. Experimental results on hyper-parameter tuning are presented in § 6.3.4.

[1]The pretrained VGG-16 network requires the input to have 3 channels. To perform transfer learning, the B-mode ultrasound image needs to be duplicated and stacked to form a 3-channel tensor. By introducing Coord-Conv, the duplication of input is not necessary any more. Therefore, it does not add any computational overhead.

6.3 Experiments and Analysis

This section describes experiments to evaluate predictions made by different UPI detectors. The dataset used in all experiments is first introduced in § 6.3.1, followed by an evaluation protocol designed to reflect the authentic detection performance when facing data shortage. Hyperparameter tuning and other implementation details are presented next, along with performance analysis and discussions on several design choices.

6.3.1 Dataset

4,871 image planes from 49 placental ultrasound volumes were annotated for UPI detection (see § 3.2.4), 31 volumes of which were confirmed as having PASD by histopathology. Unlike semantic contours in natural images, a UPI is characterized by low contrast, variable shape and sometimes contaminated by signal dropout, as shown in Figure 6.1. As stated in § 3.2.4, human experts tend to rely on global context to first identify the UPI neighbourhood and then delineate it according to local cues. Due to the muscular nature of the uterus, a UPI would normally appear to be a smooth curve in placental ultrasound images, except when intensive placental invasion penetrates muscle layers and causes irreversible UPI deformation.

6.3.2 Evaluation Protocol

For a medical image analysis application with a relatively small dataset, a k-fold cross-validation is often used for performance evaluation in compensation for the lack of test data (e.g. [94, 196–198] just to name a few). However, this can lead to over-fitting in model selection and subsequent selection bias in performance evaluation [199], producing an overly optimistic performance score for all the evaluated models. To avoid this problem, this chapter adopts a different evaluation protocol for model selection and performance evaluation: a *nested* 10-fold cross-validation [199]. Figure 6.7 illustrates the protocol. In the first step, a subject-level 10-fold split is performed on the dataset to create 10 non-overlapping test splits (blue cells in Figure 6.7). In each fold, the test split consisting of image planes from 4 - 5 volumes

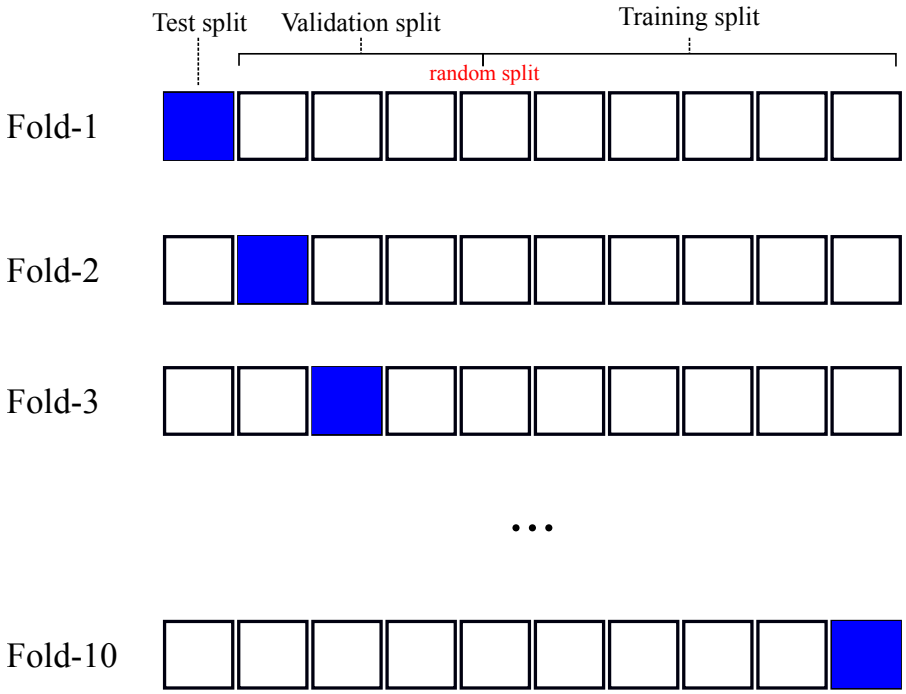


Figure 6.7: A diagrammatic demonstration of the nested cross-validation evaluation protocol.

are *held out*, while images from the remaining 44-45 volumes are further divided into train/validation splits either in a random manner or using a new k-fold split, hence the term *nested*. For simplicity, this work adopts a 80/20 subject-level random split to obtain a training split and a validation split for each fold. In the inner loop (i.e. within each fold), models are fit to the training split, whilst hyper-paramters are tuned over the validation split. In the outer loop (i.e. across folds), performance evaluation is carried out on the held-out test split. Following this protocol, for each fold, all the evaluated UPI detectors will be trained on the same training split, tuned on the same validation split and finally evaluated on the same test split. Evaluation scores on 10 test splits are reported for each UPI detector as its performance measure. As a result, this protocol will prohibit information leak from test set to training set, which otherwise would cause overly optimistic performance.

6.3.3 Evaluation Metrics

Intuitively, performance of UPI detection can be measured by standard edge detection metrics. This chapter selects two measures that are widely used in this

field [148] , namely the best F-measure on the dataset for a fixed prediction threshold (optimal dataset scale, or ODS), and the mean F-measure on the dataset for the best threshold in each image (optimal image scale, or OIS). Specifically, ODS is given by:

$$\begin{aligned} \text{ODS} &= \max\left\{\frac{2p_t r_t}{p_t + r_t} \quad | \quad t = 0 : 0.01 : 1\right\} \\ \text{OIS} &= \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \max\left\{\frac{2p_t^i r_t^i}{p_t^i + r_t^i} \quad | \quad t = 0 : 0.01 : 1\right\} \end{aligned} \tag{6.5}$$

where t denotes confidence threshold, N_{test} is the total number of images in the test set, p_t and r_t are the precision and recall at t over the entire test set, p_t^i and r_t^i are the precision and recall at t over image i. As a pre-processing step, both the UPI prediction (thresholded at t) and reference map first go through non-maximal suppression along the direction of gradient normal and then are thinned to 1-px wide via morphological thinning [200]. To compute precision and recall, UPI pixels on the reference map are bipartitely matched to their nearest predictions with a maximum tolerance distance of $\gamma\sqrt{H^2 + W^2}$ [158], i.e. the tolerance distance is proportional to the length of the image diagonal. The ratio is empirically set to $\gamma = 0.0075$ following the settings in [101, 158, 201]. The matched pixels in UPI predictions are considered true positives (TP) while the unmatched predictions are false positives (FP). False negatives (FN) refer to unmatched reference UPI pixels. Finally, precision and recall are computed according to Equation 5.7 and Equation 5.8. Both ODS and OIS consider the F-measure (also known as F-score or F_1 score), which is the harmonic mean of precision and recall. F-measure reaches a best value at 1 and worst at 0. Following [140, 148, 174], ODS is used as the primary evaluation metric since it reports the F-measure at a fixed confidence threshold.

6.3.4 Hyperparameter Tuning

In UPI-Net, GC blocks are attached to the first three convolution units (i.e. *conv-1*, *conv-2* and *conv-3*) and CGE blocks to the last two. This configuration is chosen in the hyper-parameter tuning. Intuitively, GC blocks enforce non-local dependency across low-level features while CGE blocks promote learning of high-level semantics. An optimal configuration that balances low-level and high-level

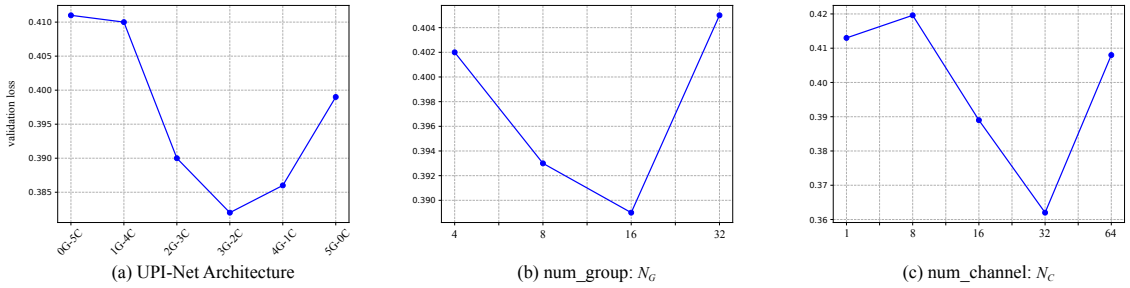


Figure 6.8: Hyper-parameter searching for UPI-Net, where an iterative strategy is applied for better efficiency.

representation learning is desired. To this end, the number of GC and CGE blocks are tuned to obtain different network variants. *mG-nC* is used to represent the first m convolution units with GC blocks attached and the last n convolution units with CGE blocks attached. For example, the proposed UPI-Net is denoted as *3G-2C*. Figure 6.8a displays the average validation losses for different GC / CGE configurations, where *3G-2C* is selected.

There are two more hyper-parameters introduced in Figure 6.4, namely the number of groups (N_G) in CGE's group convolution layer and the number of channels (N_C) in the feature aggregation layer. Similar to tuning the GC / CGE configuration, N_G and N_C are varied and tested on the validation splits. Results are summarized in Figure 6.8b-c. Note that for simplicity, the GC / CGE configuration is fixed to *3G-2C* when searching for the optimal N_G. N_G is then fixed at the optimal value when searching for the optimal N_C. Such an iterative strategy efficiently reduces the hyper-parameter searching space. As a result, $N_G = 16$ and $N_C = 32$ are selected. It is noted that setting both hyper-parameters to larger values (e.g. $N_G = 32$ and $N_C = 64$) does not necessarily reach a better performance for UPI detection.

6.3.5 Implementation Details

Parameters from an ImageNet-pretrained VGG16 are used to initialize parameters of corresponding layers in HED, CASENet, DS-FPN and the proposed UPI-Net. They are then fine-tuned on the UPI dataset. Additionally, a DCAN model is re-implemented according to the original design choice in [202] without pretraining. Convolutional layers that are not initialized by pretrained models are randomly

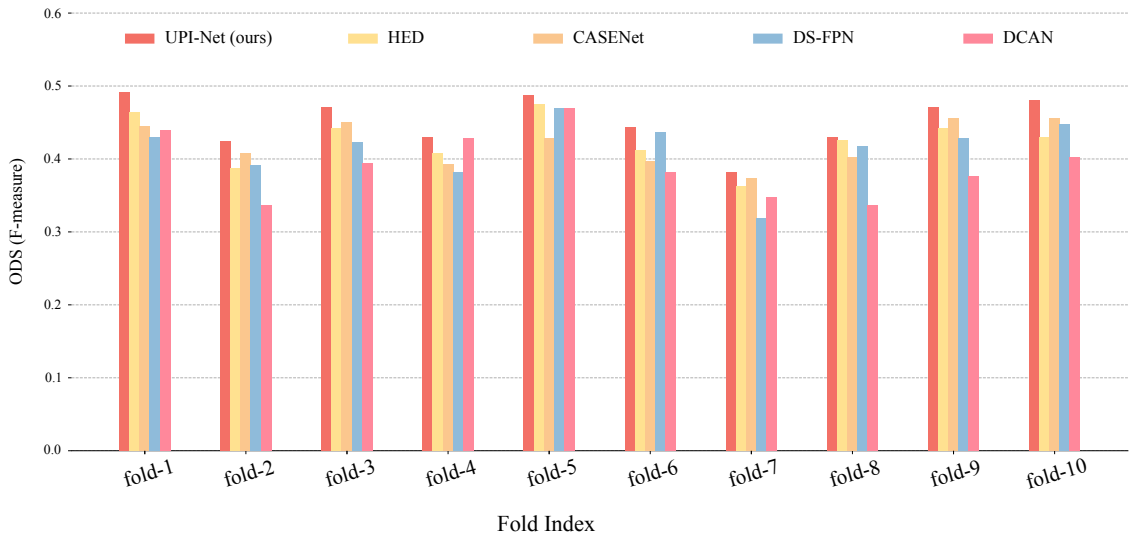


Figure 6.9: Fold-wise performance comparison among UPI detectors.

initialized with zero-mean Gaussian distributions, following the method in [177]. During training, a patch of 320×480 px is randomly cropped from the input images. The following data augmentation steps are then performed. Each side of the patch is further randomly padded (with zeros) or cropped by up to 10 percent relative to its original size. An intensity correction is applied to the patch:

$$\hat{I} = \begin{cases} I^{\gamma}, & \gamma \sim \text{uniform}(0.7, 1.3) \quad \text{with a probability of } 0.5 \\ \lambda \log_2(1+I), & \lambda \sim \text{uniform}(0.85, 1.15) \quad \text{with a probability of } 0.5 \end{cases} \tag{6.6}$$

where I is the original intensity and $\hat{I}$ is the corrected intensity.

The image intensity is zero-centered by subtracting the mean. The mini-batch size is set to 8. An AdamW optimizer is adopted with the initial learning rate of 0.0003 [74, 75]. The weight-decay is set to 0.0002. This hyper-paramater configuration is shared by all benchmark models and UPI-Net variants. All the models were implemented with PyTorch and trained for 40 epochs with early stopping on an NVIDIA DGX-1 with P100 GPUs. The early stopping has a patience level of 10, i.e. the training will stop if no ODS improvement is observed on validation split for 10 consecutive epochs.

Table 6.1: The performance of UPI detectors on the test sets in a nested 10-fold cross validation. All results are in the format of *median [first, third quartile]*. $\downarrow_0$ indicates a lower value is more appreciated, with 0 being the best in theory. $\uparrow_1$ indicates a higher value is more appreciated, with 1 being the best in theory. ODS is the primary metric.

Model	Params	FLOPs	ODS $\uparrow_1$	OIS $\uparrow_1$
HED [140]	14.7 M	52.3 G	0.427 [0.409, 0.442]	0.469 [0.445, 0.487]
CASENet [174]	14.7 M	52.3 G	0.418 [0.399, 0.449]	0.460 [0.442, 0.488]
DS-FPN [116]	15.1 M	56.2 G	0.426 [0.398, 0.435]	0.465 [0.442, 0.480]
DCAN [202]	8.6 M	12.1 G	0.388 [0.355, 0.422]	0.439 [0.407, 0.473]
UPI-Net (ours)	14.7 M	53.5 G	**0.458 [0.430, 0.479]**	**0.493 [0.474, 0.518]**

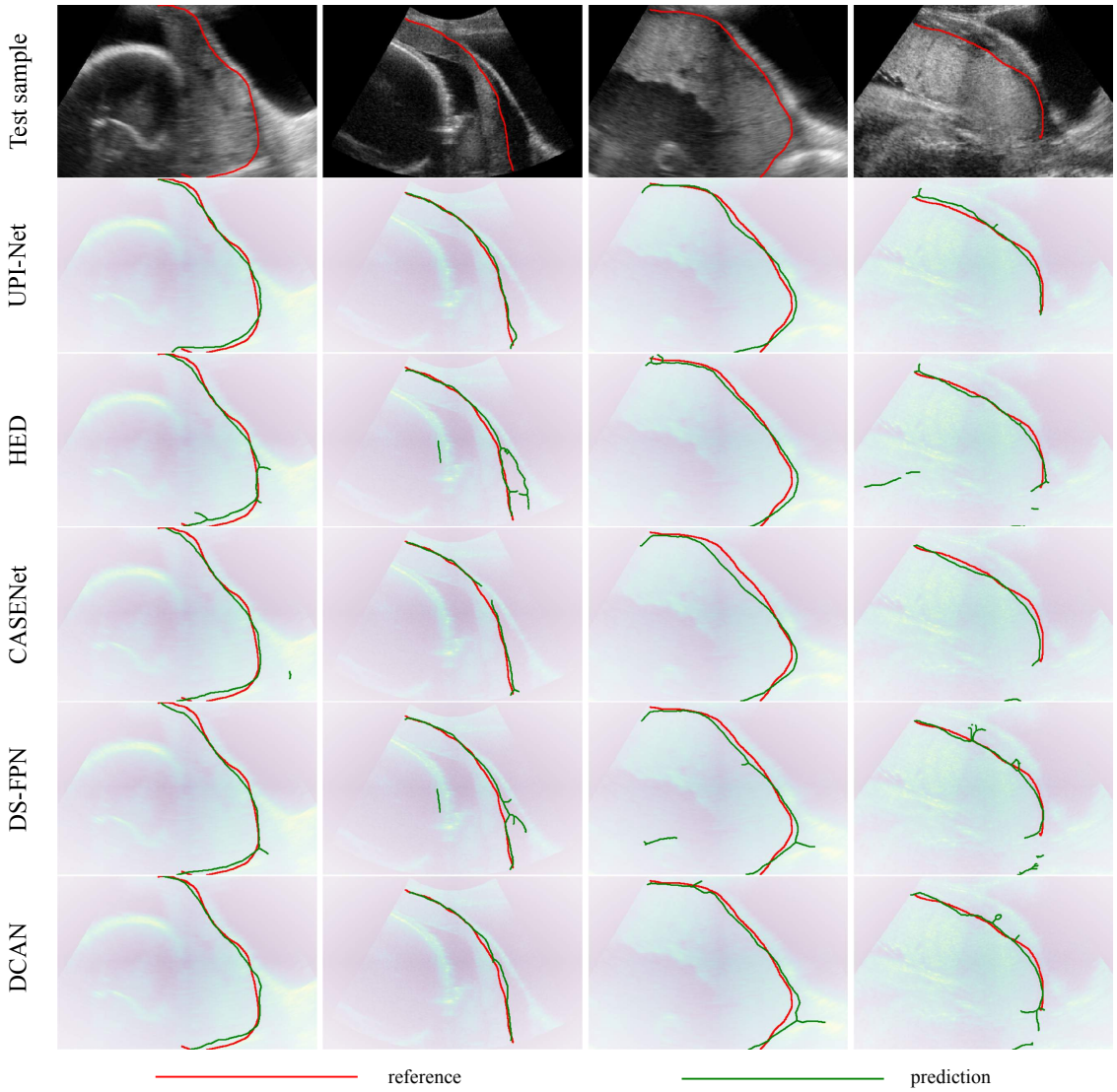


Figure 6.10: Predictions from the proposed UPI-Net model and other benchmarks. UPI-Net suppresses a number of UPI-like false positives compared with other methods.

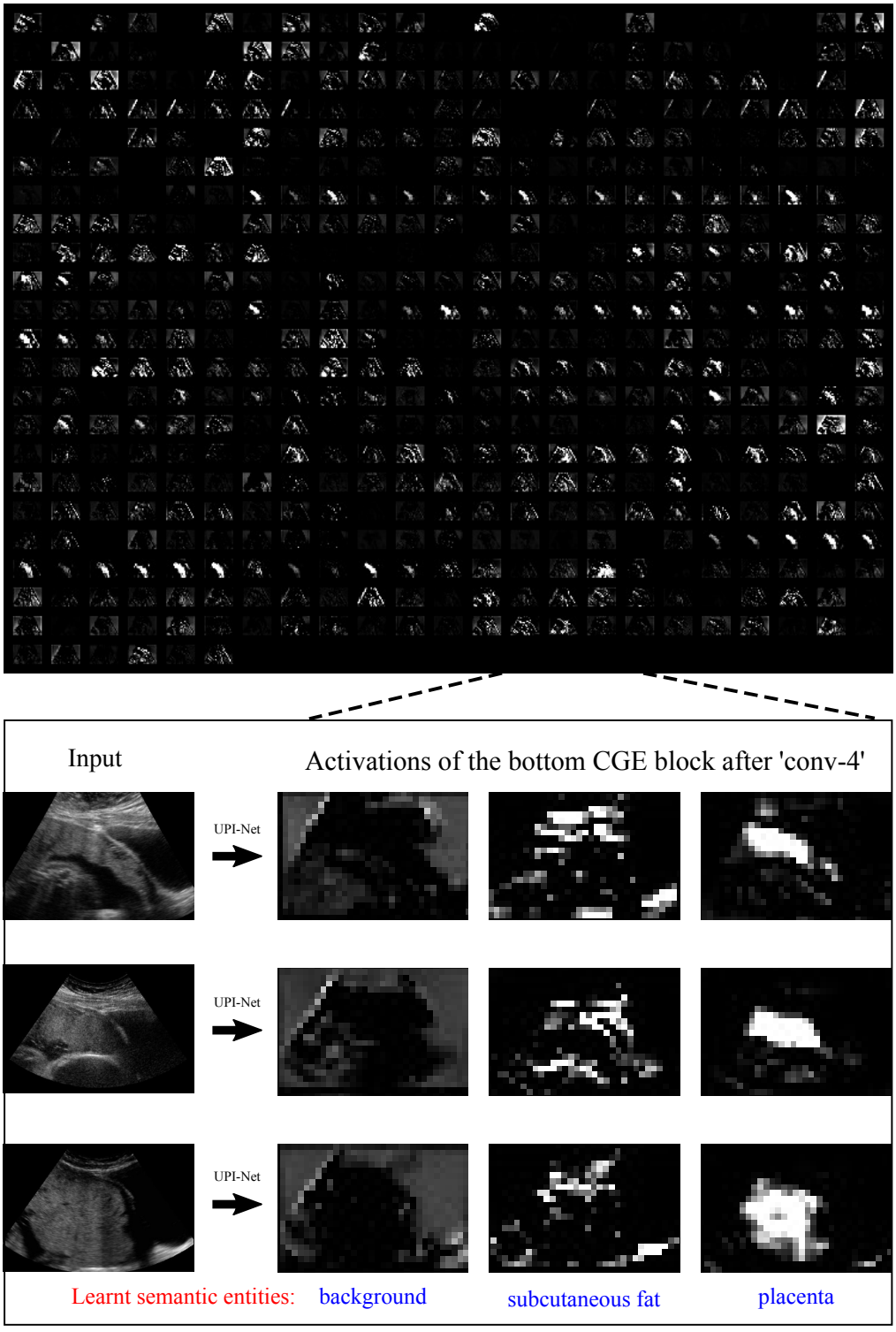


Figure 6.11: Semantic entities learnt by UPI-Net.

Table 6.2: Ablation studies on Coord-Conv layers and the side-output supervision.

Model	Coord-Conv	Side-output	ODS$\uparrow_1$
Baseline-1	✗	✓	0.438 [0.416, 0.463]
Baseline-2	✓	✗	0.444 [0.422, 0.454]
UPI-Net	✓	✓	0.458 [0.430, 0.479]

6.3.6 Results

Fold-wise detection performance of evaluated UPI detectors is illustrated in Figure 6.9. As shown in Table 6.1, median, first and third quartile of the 10-fold test results are also presented. The proposed UPI-Net outperforms four competitive benchmarks in ODS and OIS, without introducing a considerable amount of computational overhead in terms of model sizes and floating point operations. Paired t-tests confirm that ODS and OIS scores of UPI-Net are significantly higher than those from the rest benchmark models (with p-value < 0.001). Test samples with confidence threshold $t = 0.5$ are displayed in Figure 6.10. It is shown that predictions from UPI-Net are enhanced from a global perspective by suppressing unwanted UPI-like false positives and maintaining spatial smoothness of the curve (fewer false negatives).

To investigate how the Coord-Conv layer and the additional side-output supervision influence the performance of UPI detection, ablation studies are carried out by removing them respectively and re-training the corresponding baseline models. According to Table 6.2, UPI-Net benefits from both of them. Importantly, it costs no additional computational resources to add the Coord-Conv layer to the network. Although not used during testing, the side-output from *conv-5* modulates the training process to achieve better UPI detection.

It is expected that introducing CGE modules would enable the network to learn high-level semantic entities related to placental geometry more effectively thus contributes to UPI detection. As displayed in Figure 6.11, activation maps from the bottom CGE block after *conv-4* reveal some of the semantic entities learnt by UPI-Net. From 512-channel convolutional kernels, a few of them display specific semantic characteristics. Particularly, the 453^{rd} kernel appears to capture the placenta itself.

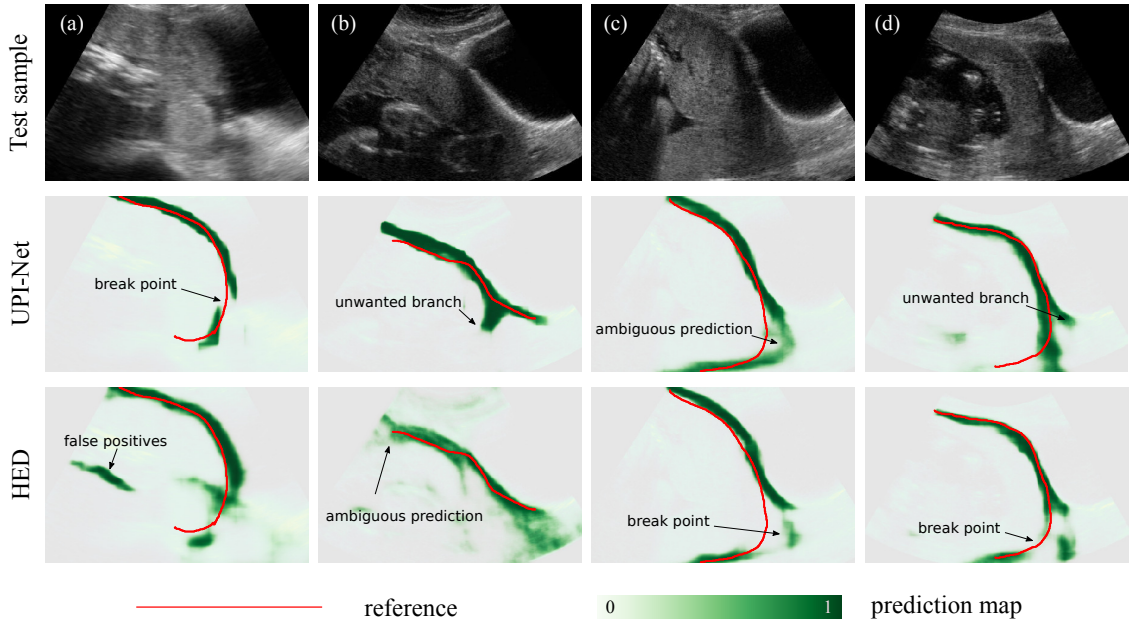


Figure 6.12: Sub-optimal predictions made by UPI-Net and HED indicates there is room to improve performance.

Note that no supervision signal associated with the placenta location is available during training. This network dissection can be further investigated to help interpret CNN's behaviours [203], with the potential to be applied in clinical settings to assist operators in better interpreting the scene by visualizing regions of interest.

6.3.7 Limitation

As a standard edge detection measure, ODS provides a tolerance mechanism, controlled by γ, in order to alleviate influence of localization ambiguity near edges. This is particularly necessary due to the indistinct appearance in UPI neighbourhood. This chapter empirically sets $\gamma = 0.0075$ following previous work [101, 158]. However, a larger γ may be more appropriate since manually localizing UPI is much harder than localizing object boundaries in natural images. Ignoring the impact of γ is one notable limitation this chapter. Identifying a well-suited γ may require redundant annotations and relevant statistical analysis, as shown in § 5.5. This shall be investigated in future work.

6.4 Conclusions

This chapter has presented a novel architecture named UPI-Net for UPI detection in placental ultrasound images, which effectively produces more plausible UPI predictions via lightweight global contextual modelling compared to competitive benchmarks. To address the challenging problem of UPI detection, this chapter proposes to formulate it as a semantic edge detection task. It was found that a bottom-up estimation of pixel-level likelihood can produce reasonable localization of the UPI neighbourhood. However, it remains difficult to accurately predict a smooth, spatially continuous and crisp UPI. False positive predictions may occur owing to the limited size of receptive fields. It was shown that incorporating global context (GC) blocks in UPI-Net introduces global contextual cues in representation learning, which can help filter out UPI-like false positive edges in the images. To enhance the formation of high-level semantics, a convolutional group-wise enhancement (CGE) block was proposed. It serves as a self-attention module that re-weights features within groups to better capture high-level semantic entities. Furthermore it was shown that convolutional kernels that capture entities such as subcutaneous fat and the placenta itself can arise without explicit supervision. It was found that UPI-Net outperformed several competitive baselines without introducing considerable computational overhead.

Figure 6.12 displays some sub-optimal predictions made by the proposed UPI-Net and the HED model. For instance, a UPI is supposed to be a continuous curve rather than a set of broken segments (Figure 6.12a,c-d). Moreover, the networks are generally not good at inferring the UPI in low-contrast regions, causing either ambiguous predictions or unwanted branches (Figure 6.12b-d). UPI detectors may benefit from more training data, additional supervision signals, and specialized objective functions that enforce geometric constraints to correct these sub-optimal behaviours. Indeed, Chapter 7 presents a method that yields a more accurate UPI detection by exploiting augmented supervision signals to further regularize network behaviours.

7

Auxiliary Learning for Utero-placental Interface Detection

Contents

7.1 Background and Motivation 120
7.2 Exploiting Auxiliary Tasks for Regularization 121
7.2.1 Gaussian Density Map and Adversarial Loss 121
7.2.2 Gradient-sensitive Mask and Spatial Focus 126
7.2.3 UPI-Net++: Architecture and Optimization 132
7.3 Experiments and Analysis 134
7.3.1 Dataset and Evaluation Protocol 134
7.3.2 Implementation Details 135
7.3.3 Performance Analysis 137
7.3.4 Limitations 142
7.4 Conclusions 142

Chapter 6 presented UPI-Net, a UPI detection model that incorporates global context modelling and semantics enhancement blocks into a fully convolutional network [145]. While the approach can perform effective UPI detection, the improvement over previous work is marginal. With UPI binary masks being the only source of supervision, the cross-entropy objective function (Equation 6.1) is not good at outputting UPI predictions that are both accurate and crisp [163], as shown in Figure 6.12. Without strong regularization of network parameters [204], it can be inefficient to rely on architectural adjustment alone (e.g. self-attention

modules [180], skip connection [90]) to fix those sub-optimal behaviours listed in Figure 6.12, especially when there is a limited amount of data available. This chapter further addresses the sub-optimal behaviours of UPI-Net. Methods are presented to achieve more accurate UPI detection in terms of ODS by constructing auxiliary tasks to enforce stronger network regularization.

7.1 Background and Motivation

So far, expert annotations of UPI are solely used to construct UPI binary masks for training. It is intuitive to consider auxiliary tasks that are beneficial to UPI detection without requiring any extra annotation. In the field of machine learning, *auxiliary learning* refers to the learning strategy to construct auxiliary tasks that contribute to solving primary tasks [205]. Note that it differs from multi-task learning, where all tasks are primary tasks. Much like multi-task learning in deep neural networks [206], the standard approach to deep auxiliary learning is to construct task-specific branches that share layers at earlier stages. For instance, Zhang et al. optimized facial landmark detection together with heterogeneous but relevant tasks: head pose estimation and facial attribute inference [207]. Valada et al. proposed to jointly learn global pose and visual odometry in order to constrain parameter searching space to obtain more consistent pose estimates [208]. Zhao et al. addressed the challenging task of image-based crowd counting by leveraging heterogeneous attributes of the ground truth density maps to generate binary masks, depth maps, and counts for auxiliary learning [209]. Shi et al. constructed segmentation maps and global density arrays derived from the ground-truth point annotations as auxiliary supervision signals to improve counting performance [81].

The premise of this chapter is that training on carefully designed auxiliary tasks can provide additional regularization that would benefit the original UPI detection task. In particular, this chapter presents methods to derive two auxiliary supervision signals from the UPI reference annotation. The first one is a Gaussian density map that models the likelihood distribution near UPI pixels. The second one is a gradient-sensitive segmentation mask that models UPI-related spatial attention on placenta

ultrasound with relaxation on low-contrast regions. Based on UPI-Net, this chapter presents several architectural refinements in order to integrate auxiliary tasks.

The contributions of this chapter are two-folds. First, two image-based transformations are presented to generate supervision signals for auxiliary tasks that are relevant to UPI detection. Second, a novel fully convolutional architecture, namely UPI-Net++, is proposed to detect UPI in placenta ultrasound images via auxiliary learning. In particular, the proposed UPI-Net++ is trained to attend task-relevant regions through supervision from the derived gradient-sensitive masks, whilst the network predictions are regularized to be both accurate and crisp, thanks to supervision from the UPI binary masks and the derived Gaussian density maps respectively. According to experiments, UPI-Net++ outperforms other competitive benchmarks in terms of edge detection evaluation metrics.

7.2 Exploiting Auxiliary Tasks for Regularization

As with Chapter 6, UPI detection is formulated as a pixel-wise binary classification problem. Let the training set be denoted as $D = \{(\mathcal{X}, \mathcal{Y})\}$, where a sample $\mathcal{X} = \{x^p, p = 1, \cdots, |\mathcal{X}|\}$ is a placental ultrasound image and $\mathcal{Y} = \{y^p, p = 1, \cdots, |\mathcal{X}|\}$ denotes the corresponding reference UPI binary mask with $y^p \in \{0, 1\}$. The overall goal is to train a network Ψ with parameters $\mathbf{W}$ to predict the probability map $\hat{\mathcal{Y}} = \{\hat{y}^p, p = 1, \cdots, |\mathcal{X}|\}$, where $\hat{y}^p$ is the network estimation of y^p at each pixel position p.

This section introduces two auxiliary tasks that help regularize Ψ to produce accurate and crisp UPI predictions. As a key pre-processing step, each pair of $(\mathcal{X}, \mathcal{Y})$ is used to derive two auxiliary supervision signals for corresponding tasks: a Gaussian density map $\mathcal{G}$ (see § 7.2.1) and a gradient-sensitive mask $\mathcal{M}$ (see § 7.2.2), as illustrated in Figure 7.1. An overview of the proposed method, UPI-Net++, is shown in Figure 7.6.

7.2.1 Gaussian Density Map and Adversarial Loss

As shown in Figure 6.12, an obvious sub-optimal behaviour of the UPI-Net and the rest benchmark models is the lack of *spatial continuity* for their outputs, i.e.

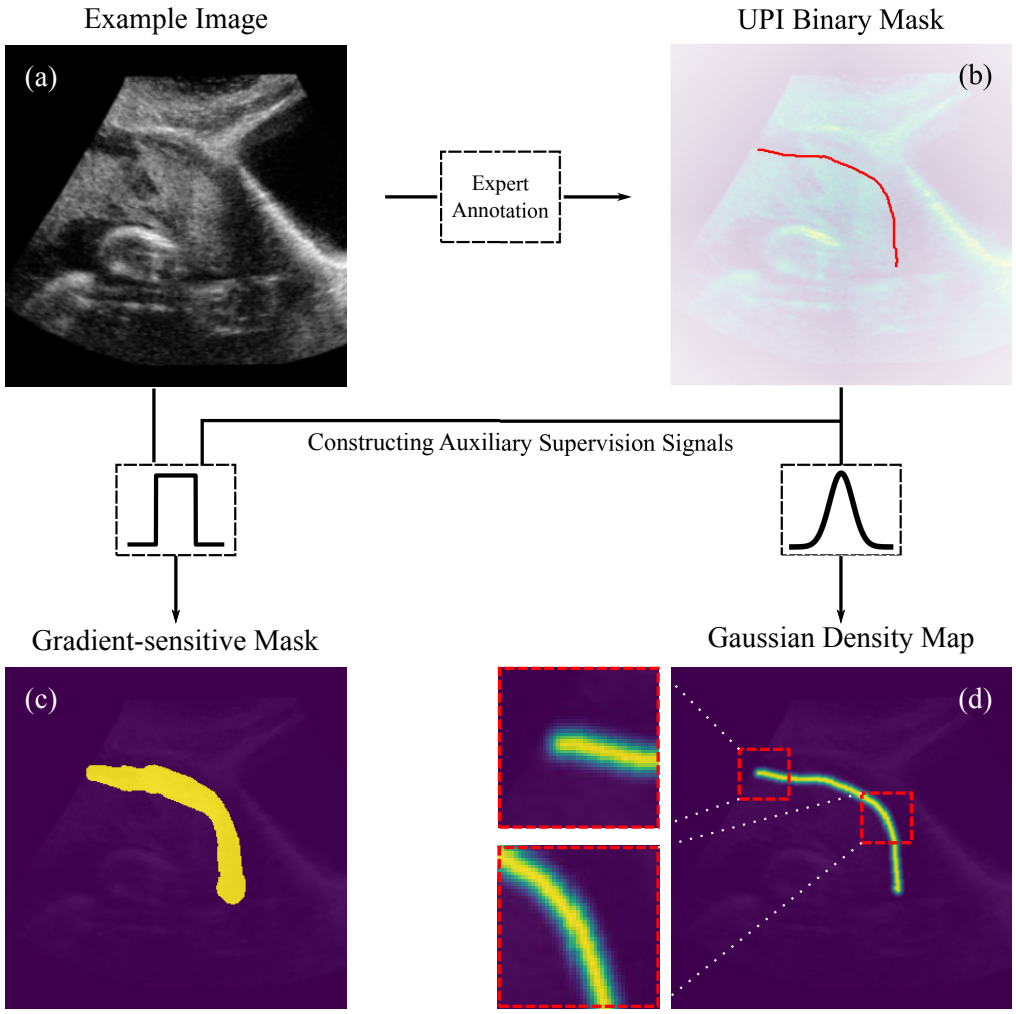


Figure 7.1: Constructing auxiliary supervision signals given an expert annotation.

UPI predictions do not form a continuous and smooth curve as it is supposed to given the muscular nature of the uterine wall. The proposed GC block is designed to incorporate global contextual cues as an effort to learn spatial correspondence among pixels with high UPI likelihoods to implicitly promote spatial continuity, which is shown to be somewhat effective with marginal improvement.

Another sub-optimal behaviour is associated with crispness [163]. In [163], crispness is measured by decreasing the maximum tolerance distance in ODS evaluation, i.e. the degree of precisely localizing edge pixels. While the examined UPI detectors have achieved reasonably high ODS scores, the quality of the prediction map is less satisfactory. For instance, Figure 7.2 displays side-outputs of the HED [140], where the UPI likelihood (e.g. Side-4 and Side-5) are blurry and

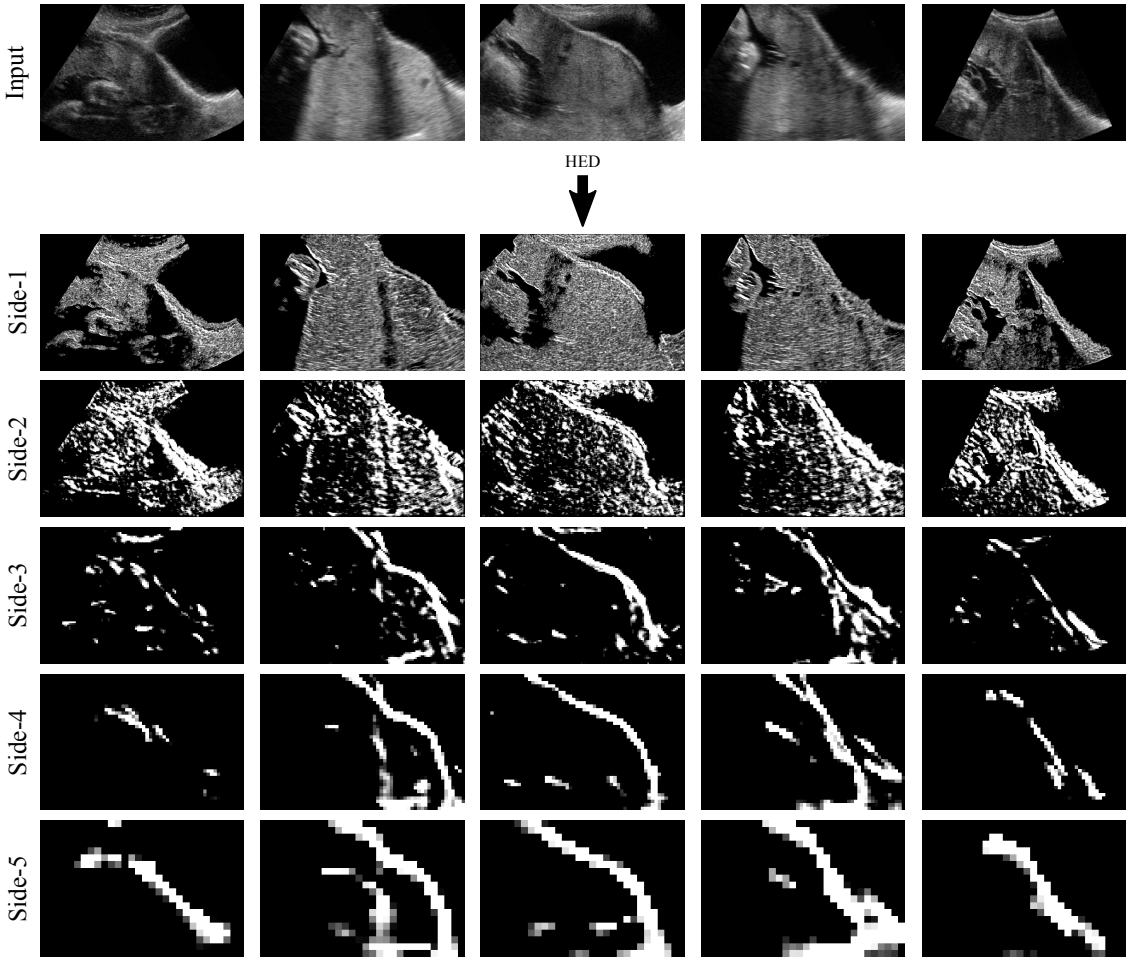


Figure 7.2: Side-outputs from HED [140].

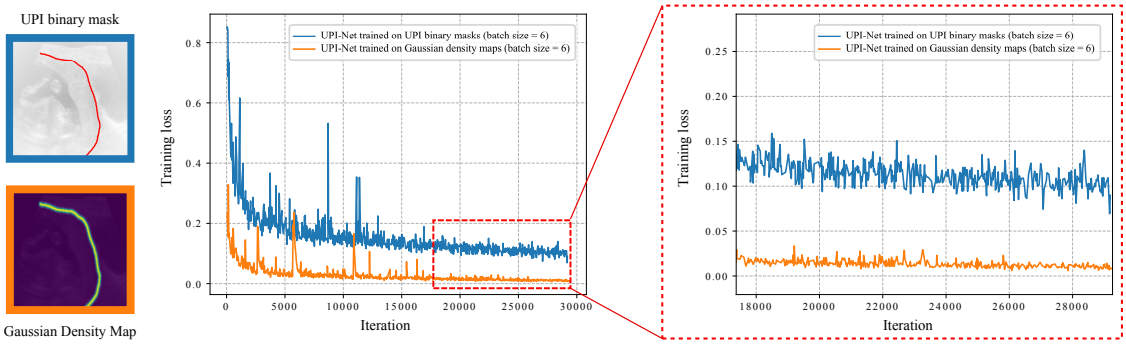


Figure 7.3: Loss comparison.

wide in the UPI neighbourhood, i.e. do not match well with the actual UPI. One reason is that the low spatial resolution at final stages cause the network to output coarse predictions that cannot be properly refined via bilinear upsampling. It is shown in Figure 7.3 that the UPI-Net trained on binary masks display stronger

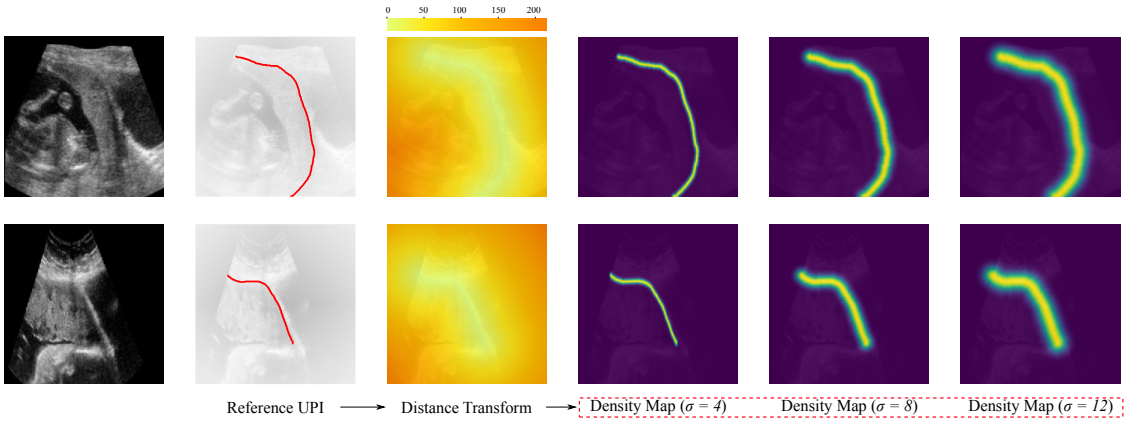


Figure 7.4: An illustration of the process to generate Gaussian density maps. The hyper-parameter σ can be tuned to obtain maps of different scales.

fluctuation in training losses even after global convergence compared to the same model trained on Gaussian density maps, which will be covered shortly. The fluctuation indicates that models trained on binary masks are more sensitive to parameter updates thus are more difficult to optimize in practice. Intuitively, the predicted likelihood $Pr(\hat{y} = 1|\mathcal{X}; \mathbf{W})$ is expected to display *Gaussian behaviour* along the UPI normal direction, where the more likely pixels are centred among those less likely, as shown in Figure 7.1d.

To overcome these limitations, this chapter presents a method to derive a Gaussian density map from the UPI binary mask and exploit this map as an auxiliary supervision signal to enforce spatial continuity and Gaussian behaviour via an adversarial loss.

Gaussian Density Map Generation

A Gaussian density map $\mathcal{G}_\sigma$ is obtained as a function of a UPI binary mask $\mathcal{Y}$ and a pre-defined standard deviation term σ. In the first step, a distance transform is performed on $\mathcal{Y}$ to get a distance map $\mathcal{D}$. At each pixel position p, $\mathcal{D}^p$ is the distance from p to its nearest UPI pixel. Obviously, $\mathcal{D}^p = 0$ for all UPI pixels. Accordingly, the Gaussian density map for each pixel position p is determined as:

$$\mathcal{G}_\sigma^p = \begin{cases} \exp\left(-\frac{\mathcal{D}^p}{2\sigma^2}\right) & \text{if } \mathcal{D}^p < 3\sigma \\ 0 & \text{otherwise} \end{cases} \tag{7.1}$$

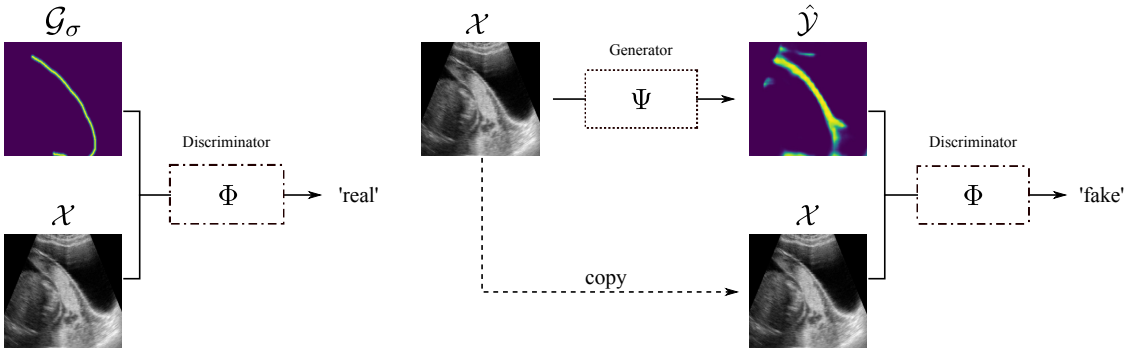


Figure 7.5: An illustration of training a conditional generative adversarial network (cGAN) to produce UPI predictions that look like Gaussian density maps. The discriminator Φ learns to distinguish fake inputs (predictions from the generator conditioned on input images) from real inputs (Gaussian density maps conditioned on input images), while the generator Ψ learns to fool the discriminator.

This generation process is illustrated in Figure 7.4, with $\sigma = 4$, $\sigma = 8$ and $\sigma = 12$ respectively. In practice, the value of σ is determined by hyper-parameter searching on the validation split.

Adversarial Regularization

In order to produce UPI predictions that are both accurate and crisp as well as to enforce spatial continuity without sacrificing detection performance, an adversarial loss is integrated into UPI-Net++ [77, 210]. In general, the corresponding auxiliary task is to make the UPI prediction look like a Gaussian density map conditioned on the input image. In the layout of a conditional generative adversarial network (cGAN), the UPI detector Ψ is considered a *generator* that is trained to produce a UPI prediction $\hat{\mathcal{Y}}$ in a feed-forward pass given a placental ultrasound image $\mathcal{X}$ such that $\hat{\mathcal{Y}}$ cannot be distinguished from 'real' Gaussian density maps by an adversarially trained discriminator Φ, which is trained to detect the generator's 'fake' outputs. The training procedure of a cGAN is diagrammed in Figure 7.5. Overall, the objective of this auxiliary task is expressed as:

$$L_{adv}(\Psi, \Phi) = \mathbb{E}_{\mathcal{X},\mathcal{G}_\sigma}[\log \Phi(\mathcal{X}, \mathcal{G}_\sigma)] + \mathbb{E}_{\mathcal{X},\hat{\mathcal{Y}}}[\log(1 - \Phi(\mathcal{X}, \hat{\mathcal{Y}}))] \tag{7.2}$$

where $\hat{\mathcal{Y}}$ is obtained by a sigmoid function of the generator's output $\Psi(\mathcal{X})$:

$$\hat{\mathcal{Y}} = \frac{1}{1 + \exp(-\Psi(\mathcal{X}))} \tag{7.3}$$

Table 7.1: Architecture of the discriminator network Φ

Layer	Depth	Kernel	Stride	Padding	Activation
Convolution	2→64	4×4	2	1	Leaky ReLU
Convolution	64→128	4×4	2	1	IN+Leaky ReLU
Convolution	128→128	4×4	2	1	IN+Leaky ReLU
Convolution	128→128	4×4	2	1	IN+Leaky ReLU
Convolution	128→256	4×4	2	1	IN+Leaky ReLU
Average Pooling	-	-	-	-	-
Fully Connected	256→128	-	-	-	Tanh
Dropout ($p = 0.2$)	-	-	-	-	-
Fully Connected	128→64	-	-	-	Tanh
Dropout ($p = 0.2$)	-	-	-	-	-
Fully Connected	64→1	-	-	-	Sigmoid

In Equation 7.2, Ψ aims to minimize L_{adv} against the adversarial discriminator Φ, which aims to maximize L_{adv}. Therefore, an optimal generator is given by:

$$\Psi^* = \arg\min_{\Psi}\max_{\Phi} L_{adv}(\Psi, \Phi) \tag{7.4}$$

Ψ is essentially a binary classification network that takes the channel-wise concatenation of $(\mathcal{X}, \hat{\mathcal{Y}})$ as well as $(\mathcal{X}, \mathcal{G}_\sigma)$. This chapter constructs Ψ by following the architectural design choices in [211]. Five 4×4 convolutional layers with stride of 2 and three fully connected layers are stacked with interleaved instance normalization (IN) layers [67] and leaky ReLU activation layers [65]. Two dropout layers ($p = 0.2$) are used to reduce overfitting. The configuration of Ψ is displayed in Table 7.1.

Joint optimization of Ψ and Φ is achieved by alternating between gradient descent steps of the two networks (see § 7.2.3). As shown in Figure 7.6, L_{adv} is computed alongside the UPI prediction loss L_{upi}, which is the weighted cross-entropy loss between $\hat{\mathcal{Y}}$ and $\mathcal{Y}$ given by Equation 6.1. To reduce memory footprint, both $\hat{\mathcal{Y}}$ and $\mathcal{Y}$ are downsampled by 4 before being fed into Φ.

7.2.2 Gradient-sensitive Mask and Spatial Focus

Failure cases such as false positive predictions, low-confidence predictions and unwanted branches (Figure 6.12) can be attributed to lack of spatial focus, i.e. the network is not good at identifying and attending regions related to UPI. Intuitively, pixels that are not within such regions should be filtered out immediately.

From the perspective of network optimization, it is also desirable to disregard irrelevant pixels in order to prevent loss overwhelming since UPI detection is extremely imbalanced in terms of positive/negative samples. To overcome this issue, UPI-Net++ integrates an auxiliary segmentation task to explicitly enforce spatial attention for UPI detection via a gradient-sensitive mask, which is an auxiliary supervision signal derived from the original binary mask by taking into consideration local gradient magnitude.

Gradient-sensitive Mask Generation

As illustrated by the red dotted bounding boxes in Figure 7.7, it is quite challenging even for experienced sonographers to identify UPI in the low-contrast regions. Intuitively, more spatial attention, reflected by width of an attention mask in UPI normal direction, should be focused on the low-contrast regions compared to the high-contrast ones. Based on this idea, this chapter proposes a method to generate a gradient-sensitive mask $\mathcal{M}$ as a function of the input image $\mathcal{X}$ and its UPI binary mask $\mathcal{Y}$, which is demonstrated in the following steps:

1. **First-order gradient approximation**. The classic Sobel kernels are fast to compute. They are convolved with the input image $\mathcal{X}$ to calculate approximations of the image gradients horizontally and vertically. Let G_h and G_v denote two images containing the horizontal and vertical gradient approximations of $\mathcal{X}$ respectively. The computations are given by:

$$G_h = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathcal{X} \quad \text{and} \quad G_v = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * \mathcal{X} \tag{7.5}$$

where $*$ denotes the two-dimensional convolution operation. G_h and G_v are combined to give the gradient magnitude G:

$$G = \sqrt{G_h^2 + G_v^2} \tag{7.6}$$

To reduce the influence of noise and artefacts, a median filtering of 10-px radius is used on G, rendering a smoothed gradient magnitude image $\tilde{G}$, as

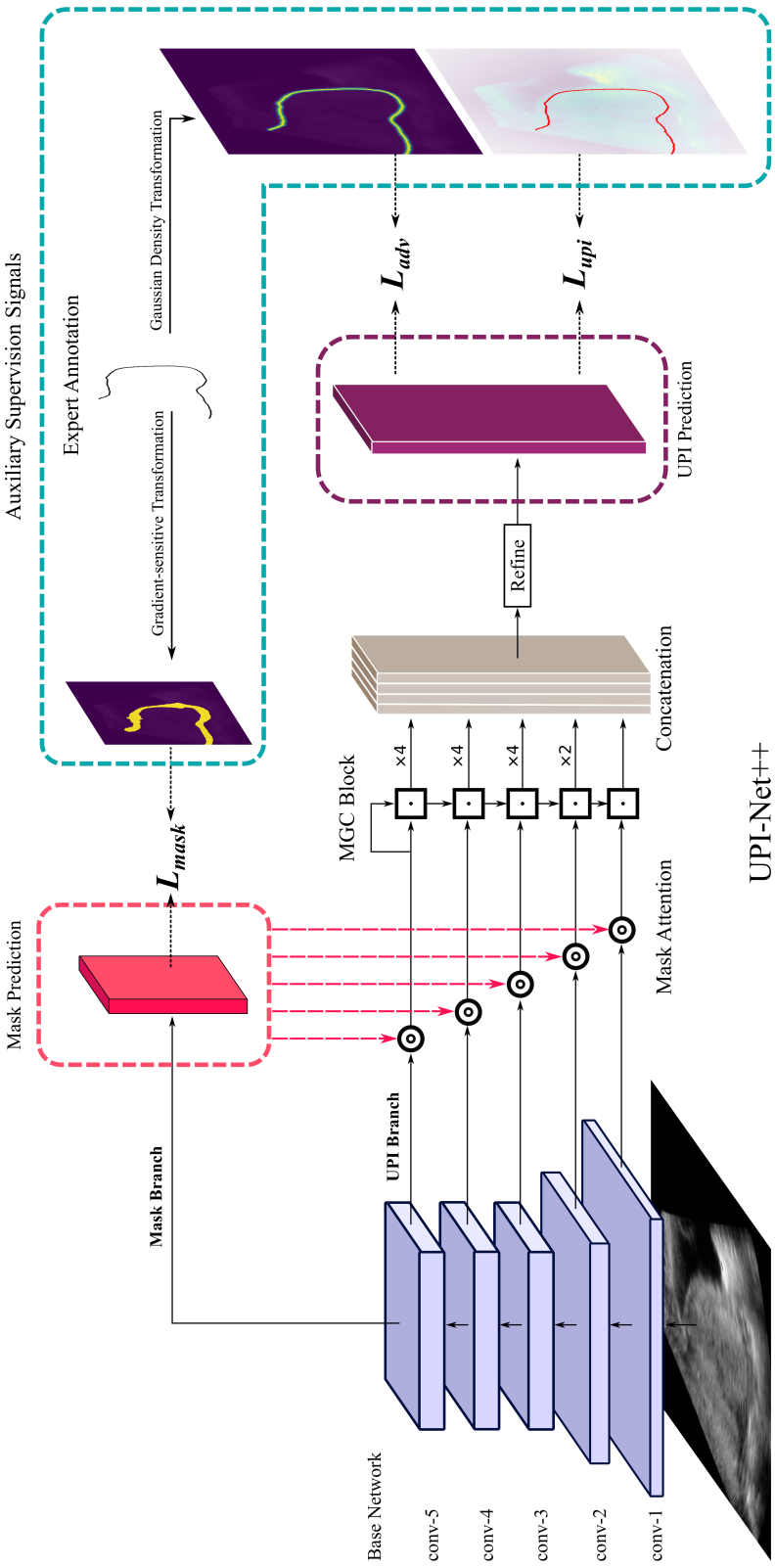


Figure 7.6: UPI-Net++.

illustrated in the third column of Figure 7.7. Intuitively, low-contrast regions correspond to small gradient magnitudes.

2. **Gradient magnitude clustering**. In this step, a distance transform is first applied to the UPI binary mask $\mathcal{Y}$ to obtain a tuple (D^p, I^p) for each pixel position p, where D^p is the Euclidean distance from p to its nearest UPI pixel and I^p is the spatial index of its nearest UPI pixel. An indexing inference is then performed to generate a gradient magnitude clustering image $\tilde{G}_c$. For each pixel position p, $\tilde{G}_c^p$ takes the gradient magnitude of its nearest UPI pixel, i.e.

$$\tilde{G}_c^p = \tilde{G}^{I^p} \tag{7.7}$$

As illustrated in the fourth column of Figure 7.7, $\tilde{G}_c$ can be used to re-weight D_p in terms of local contrast.

3. **Gradient-sensitive masking**. Much like the generation process of Gaussian density maps (Equation 7.1), a Gaussian function is first used to measure pixel-wise proximity to UPI based on a weighted Euclidean distance. The value for each pixel position p in the resulting map $\mathcal{G}_{gs}$ is determined as:

$$\mathcal{G}_{gs}^p = \exp\left(\frac{-D_p^2 \tilde{G}_c^p}{2\sigma_{gs}^2}\right) \tag{7.8}$$

where σ_{gs} is empirically set to 0.8 to control the fall-off. Note that $\mathcal{G}_{gs} \propto \exp(\tilde{G}_c)$, which indicates that a larger degree of proximity relaxation is given to low-contrast regions in order to obtain more spatial focus, compared to high-contrast regions. Finally, a binary thresholding at 0.5 produces the gradient-sensitive mask:

$$\mathcal{M}^p = \begin{cases} 1 & \text{if} \quad \mathcal{G}_{gs}^p > 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{7.9}$$

Samples are displayed in the final column of Figure 7.7, which demonstrates that gradient-sensitive masks put more spatial focus on low-contrast regions. Please refer to Figure 7.8 for more samples of Gaussian density maps ($\sigma = 4$) and gradient-sensitive masks.

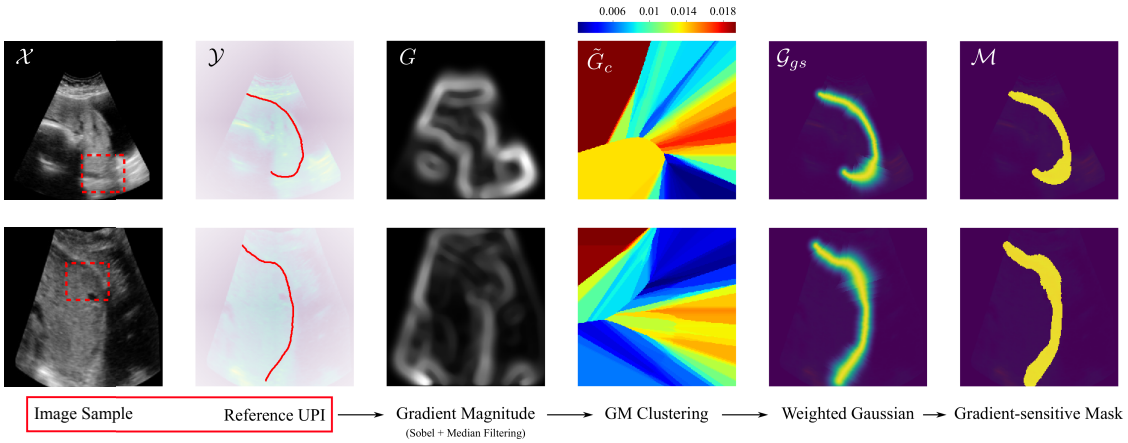


Figure 7.7: Gradient-sensitive Masking.

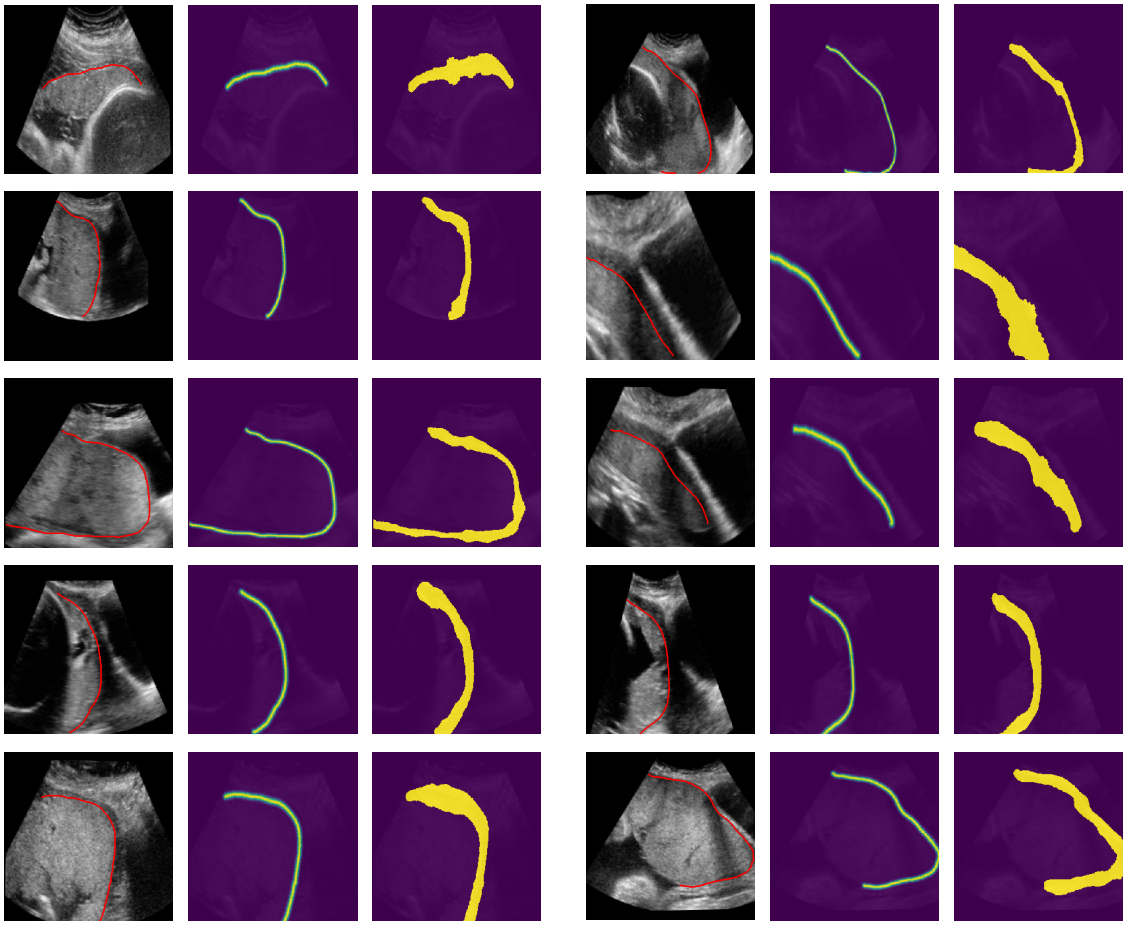


Figure 7.8: Supervision augmentation samples.

Mask Attention

A gradient-sensitive mask $\mathcal{M}$ serves as a supervision signal for the auxiliary task of segmenting UPI-related spatial focus. As shown in Figure 7.6, a mask branch

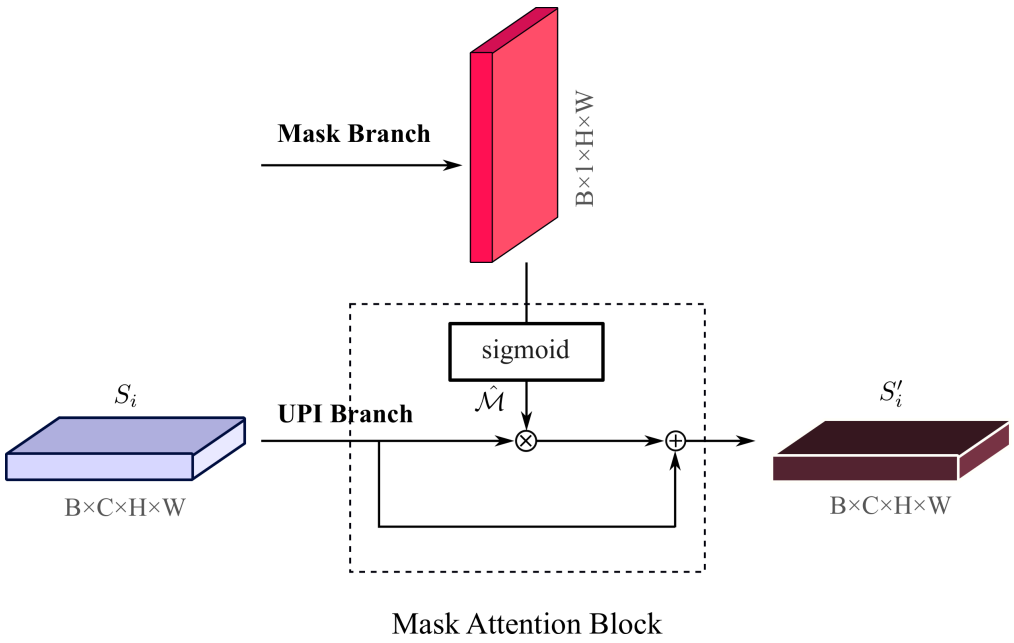


Figure 7.9: Masking attention block.

is added on top of the base network with a sigmoid activation to produce $\hat{\mathcal{M}}$, the network's estimation of $\mathcal{M}$. The objective of this auxiliary task is given by a weighted cross-entropy function:

$$L_{mask}(\hat{\mathcal{M}}; \Psi) = -\frac{\omega_M}{|\mathcal{M}|} \sum_{p \in |\mathcal{M}_+|} \log(\hat{\mathcal{M}}^p) - \frac{1}{|\mathcal{M}|} \sum_{p \in |\mathcal{M}_-|} \log(1 - \hat{\mathcal{M}}^p) \tag{7.10}$$

where ω_M is set to $\frac{|\mathcal{M}_-|}{|\mathcal{M}_+|}$, with $|\mathcal{M}_+|$ and $|\mathcal{M}_-|$ being the number of positives (i.e. the value 1) and negatives (i.e. the value 0) of $\mathcal{M}$. Similar to § 7.2.1, $\mathcal{M}$ is downsampled by 4 to reduce memory footprint.

To emphasize UPI-related spatial focus as well as filter out irrelevant features, a mask attention module is proposed to integrate $\hat{\mathcal{M}}$ into the UPI branch. Let S_i denote the i-th scale activated feature map. As demonstrated in Figure 7.9, the mask attention modules acts as a pixel-level gating mechanism. A residual structure is adopted to maintain global semantics. As a result, the mask attention module outputs a refined feature map S_i', which is given by:

$$S_i' = S_i \otimes \hat{\mathcal{M}} \oplus S_i \tag{7.11}$$

where $\otimes$ and $\oplus$ denotes channel-wise broadcasting multiplication and element-wise addition respectively. There are five scales in UPI-Net++'s hierarchy, which leads

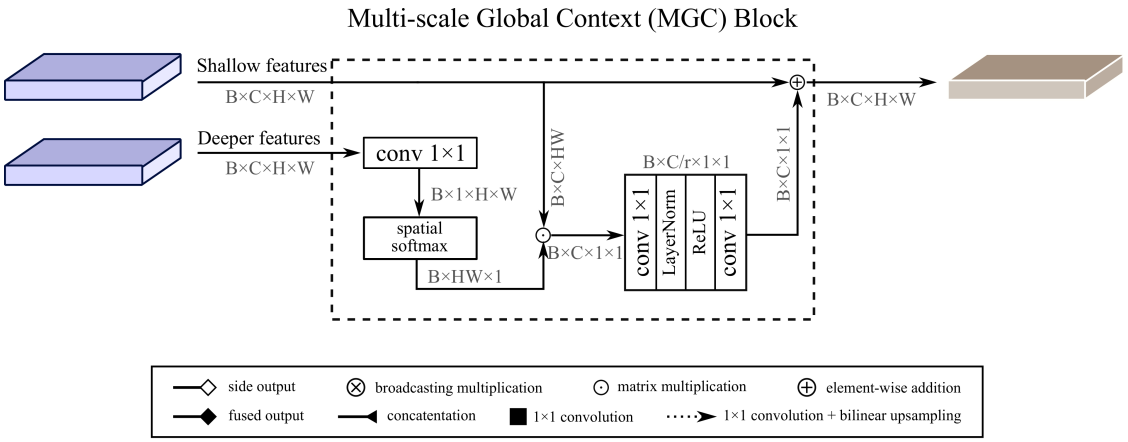


Figure 7.10: Multi-scale global context (MGC) block.

to five mask attention modules, as shown in Figure 7.6. The spatial resolution mismatches between $\{S_i\}$ and $\hat{\mathcal{M}}$ are resolved by bilinear upsampling on $\hat{\mathcal{M}}$.

Based on this formulation, the mask attention module is expected to highlight UPI-related features via spatial attention operation thus improve UPI detection performance. It should also facilitate the learning of gradient-sensitive masks through back-propagation.

7.2.3 UPI-Net++: Architecture and Optimization

As shown in Figure 7.6, UPI-Net++ contains two branches, a mask branch for gradient-sensitive mask segmentation and a UPI branch for UPI detection and adversarial regularization. To maintain spatial resolution without reducing receptive fields for UPI detection, UPI-Net++ adopts dilated convolutions in the last three network stages, which has been successfully applied in a variety of image recognition applications [178, 179]. Accordingly, the effective strides of UPI-Net++'s five scales are $\{1, 2, 4, 4, 4\}$.

Chapter 6 presented a GC block to refine network representation via a simplified self attention operation [181]. The global contextual cues come from the representation itself through a 1×1 convolution followed by a spatial softmax operation. Intuitively, representation from deeper stages is likely to contain more informative contextual cues, which leads to an extension of a GC block by incorporating multi-scale representation. Specifically, a multi-scale global context (MGC) block is

proposed in this section to further enhance context modelling. The MGC block takes two inputs: the representation at the i-th scale $\mathcal{F}_i \in \mathbb{R}^{B\times C\times H\times W}$ and the representation at a deeper scale $\mathcal{F}_{i+1} \in \mathbb{R}^{B\times C\times H\times W}$. As illustrated in Figure 7.10, the key difference between a GC block and an MGC block lies in the global attention pooling operation. Instead of using $\mathcal{F}_i$ to extract global context, an MGC block exploits the deeper representation $\mathcal{F}_{i+1}$. In UPI-Net++ (Figure 7.6), MGC blocks are integrated after mask attention modules from deeper stages to shallower stages. In the deepest stage, a MGC block is equivalent to a GC block since the global attention comes from itself. All the channel mismatches are resolved by 1×1 convolution and resolution mismatches by bilinear upsampling.

After going through MGC blocks, feature maps from different scales are linearly transformed via 1×1 convolutions to have $N_C = 32$ channels. They are then upsampled to the input's spatial resolution via bilinear interpolation and aggregated via channel-wise concatenation to obtain a 160-channel feature representation, which is then fed into a refinement block consisting of two basic residual blocks from [84] and a 1×1 convolutional layer to obtain the final UPI prediction. The overall objective function combines three objectives from the primary task and two auxiliary tasks:

$$L_{all}(\Psi, \Phi) = L_{upi}(\Psi) + \lambda_{adv} L_{adv}(\Psi, \Phi) + \lambda_{mask} L_{mask}(\Psi) \tag{7.12}$$

where $(\lambda_{adv}, \lambda_{mask})$ denote the weighting parameters of each term. Their values are determined in hyper-parameter tuning (see § 7.3.2). The adversarial term in Equation 7.12 is optimized in a minimax game with the final objective: $\Psi^* = \arg\min_\Psi \max_\Phi L_{all}(\Psi, \Phi)$. Optimization is achieved by alternating gradient descent step on Ψ and Φ. As suggested in the original GAN paper [77], rather than optimizing Ψ to minimize $\log(1 - \Phi(\mathcal{X}, \hat{\mathcal{Y}}))$ with $\hat{\mathcal{Y}} = \text{sigmoid}(\Psi(\mathcal{X}))$, the training process instead maximizes $\log(\Phi(\mathcal{X}, \hat{\mathcal{Y}}))$. At inference time, the generator Ψ (i.e. UPI-Net++) is treated as an independent UPI detector, just like UPI-Net in Chapter 6.

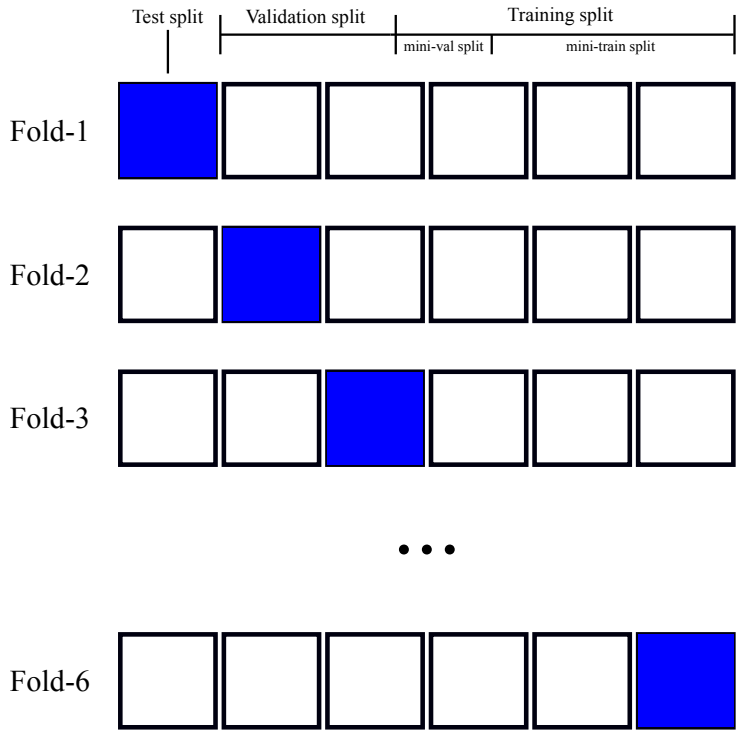


Figure 7.11: A diagram of the 6-fold nested cross-validation evaluation protocol.

7.3 Experiments and Analysis

This section describes experiments to evaluate detection performance of UPI-Net++ and compare it against several competitive benchmarks. First, the dataset and evaluation protocols used in all experiments are introduced. Then, implementation details are presented, along with performance analysis and ablation studies.

7.3.1 Dataset and Evaluation Protocol

In the Oxford-Hackensack database of PASD, 11,166 image planes from 101 placental ultrasound volumes had UPI annotations, among which 48 were confirmed with PASD by histopathology (18 AAP and 30 AIP, see § 3.2). Based on the evaluation protocol introduced in § 6.3.2, a subject-level 6-fold nested cross-validation is used to evaluate UPI detectors' performance in this chapter. As illustrated in Figure 7.11, each fold has a held-out test split consisting of image planes from 16-17 volumes. The rest image planes are divided into a training split and a validation split according to a random 80/20 subject-level split. The subject-level

split ensures that models are never trained and tested on image planes obtained from the same placental ultrasound volume. Furthermore, another random 80/20 subject-level split is carried out on the training split, resulting in a *mini-train* split and a *mini-val* split. They are constructed for hyper-parameter tuning only. As a result, a hyper-parameter configuration would be trained on the mini-train split. Model selection would be performed on the mini-val split, and finally its performance would be tested on the validation split. This way guarantees there is no information leak towards the held-out test split. Following this protocol, for each fold, all the evaluated UPI detectors will be trained on the training split, tuned on the validation split and finally evaluated on the test split. Evaluation scores on 6 non-overlapping test splits are reported for each UPI detector. Following § 6.3.3, the median, first and third quartiles of the ODS and OIS scores across folds are reported. ODS is used as the primary metric.

7.3.2 Implementation Details

Preprocessing. All placental ultrasound images in the database are rescaled to $[0, 1]$ and zero-centred by subtracting the mean. During training, a patch of 320×320 px is randomly cropped. A series of data augmentation operations are conducted:

- Each side of the patch is padded or cropped by a randomly sampled ratio between 0 and 10% relative to its original size and resized to 320×320 px.
- Each patch is translated along x-axis and y-axis by a randomly sampled ratio between 0 and 10% relative to its original size and resized to 320×320 px.
- Each patch is rotated by a randomly sampled degree between -5° to 5°.
- Gamma correction and logarithm correction are performed following Equation 6.6.

Hyper-parameters. The hyper-parameter configuration of training UPI-Net++ is as follows. The mini-batch size is set to 6. Group normalization is used between a convolutional layer and its ReLU activation [70]. Two AdamW optimizers are

adopted by Ψ and Φ respectively with the initial learning rate of 0.0003 [74, 75]. The weight decay is set to 0.0002. Similar to Chapter 6, all evaluated models were implemented with PyTorch and trained for 40 epochs with early stopping on an NVIDIA DGX-1 with P100 GPUs. Early stopping is performed with a patience level of 10, i.e. the training will stop if no ODS improvement is observed on validation split for 10 consecutive epochs. This configuration is shared by all benchmark models and UPI-Net++ variants. After hyper-parameter tuning on validation splits, the following parameters are selected for UPI-Net++. The standard deviation of Gaussian density maps is set to $\sigma = 6$. The weighting parameter λ_{mask} in Equation 7.12 is set to 0.4. To prevent L_{adv} from overwhelming the primary objective L_{upi}, an annealing strategy is applied to the weighting parameter λ_{adv}, which is obtained as a function of the epoch index $\epsilon \in \{0, 1, 2, \cdots, 39\}$:

$$\lambda_{adv}(\epsilon) = a \exp(b\epsilon) \tag{7.13}$$

where a is set to 0.05 and b is set to -0.1609. At epoch 0, λ_{adv} is initialized to 0.05. After 10 epochs, λ_{adv} will be reduced to 0.01 and then 0.002 after 20 epochs and so on. Empirically, this annealing strategy leads to higher ODS scores compared to a fixed λ_{adv}.

Base Network. As shown in Figure 7.6, there are five stages in UPI-Net++'s hierarchical architecture. Depending on the choice of base networks, feature maps may have different channel numbers at a given stage. Two base networks are investigated in this chapter: (i) VGG-16 [83], pretrained on ImageNet (stage-wise channel number: {64, 128, 256, 512, 512}); (ii) DRN-D-38 [178], pretrained on ImageNet (stage-wise channel number: {32, 64, 128, 256, 512}). The fully connected layers in these base networks are removed. The rest of convolutional layers in UPI-Net++ that are not initialized by pretrained models are randomly initialized with zero-mean Gaussian distributions, following the method in [177].

Table 7.2: Effects of σ in terms of ODS and OIS on validation splits. The base network is an ImageNet pretrained VGG-16. All results are in the format of *median [first, third quartile]* over six folds.

UPI-Net++ (based on VGG-16)	ODS$\uparrow_1$	OIS$\uparrow_1$
$\sigma = 4$	0.562 [0.559, 0.569]	0.601 [0.595, 0.605]
$\sigma = 6$	0.575 [0.563, 0.595]	0.615 [0.594, 0.631]
$\sigma = 8$	0.567 [0.562, 0.575]	0.608 [0.604, 0.613]
$\sigma = 10$	0.566 [0.562, 0.569]	0.608 [0.606, 0.610]
$\sigma = 12$	0.559 [0.552, 0.563]	0.600 [0.590, 0.610]

7.3.3 Performance Analysis

A series of data experiments were carried out to investigate UPI-Net++'s design choices and to compare its performance against competitive benchmark models. This section first analyses the effects of UPI-Net++'s key hyper-parameters and auxiliary tasks on UPI detection. Relevant models were trained on mini-train splits, tuned on mini-validation splits and tested on validation splits. It then presents comparisons to several benchmark models with respect to quantitative measures as well as qualitative illustrations.

Gaussian Density Map Standard Deviation

As shown in Figure 7.4, the standard deviation term σ in Equation 7.1 controls fall-off of the Gaussian function, which implicitly influences the fall-off of UPI predictions through the adversarial regularization by L_{adv}. Table 7.2 shows UPI detection performance of UPI-Net++ trained on Gaussian density maps with different σs in terms of ODS and OIS. $\sigma = 6$ achieves the best result on validation splits only by a small margin. The median ODS is 0.575 and the median OIS is 0.615, both slightly outperforming the rest choices of σ. This is an expected result since the adversarial term is designed to regularize the crispness of model outputs, which is not directly associated with ODS and OIS. As a result, $\sigma = 6$ achieves a desirable balance between crispness and accuracy, which is used in the rest of this chapter.

Table 7.3: Effects of MGC blocks in terms of ODS and OIS on validation splits. Two base networks of UPI-Net++, VGG-16 and DRN-D-38, are tested. All results are in the format of *median [first, third quartile]* over six folds.

Base Network	MGC	L_{adv}	L_{mask}	ODS$\uparrow_1$	OIS$\uparrow_1$
VGG-16	✗	✓	✓	0.558 [0.554, 0.578]	0.602 [0.597, 0.614]
VGG-16	✓	✓	✓	0.575 [0.563, 0.595]	0.615 [0.594, 0.631]
DRN-D-38	✗	✓	✓	0.590 [0.564, 0.599]	0.625 [0.601, 0.641]
DRN-D-38	✓	✓	✓	0.596 [0.570, 0.602]	0.631 [0.604, 0.644]

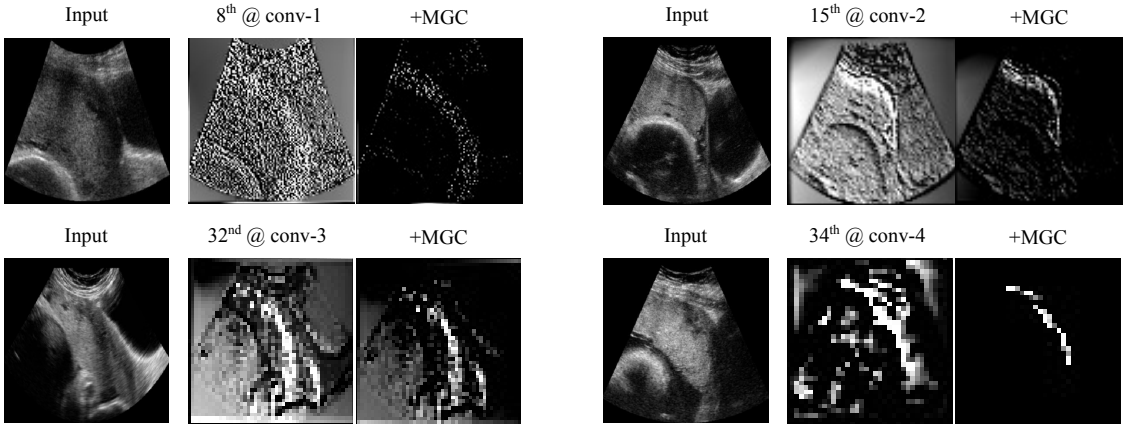


Figure 7.12: Representations before (middle column) and after (right column) fed into a MGC block. Samples from different scales are displayed.

Effect of MGC Blocks

To analyse the effect of the proposed MGC blocks, experiments are performed by replacing MGC blocks with GC blocks, i.e. cutting off global contextual cues from higher scales. This results in performance reduction in terms of ODS and OIS as shown in Table 7.3. Without MGC, the VGG-16 model has a performance decrease of 0.017 and 0.013 in terms of ODS and OIS median respectively, while the DRN-D-38 model has a performance decrease of 0.006 and 0.006 in terms of ODS and OIS median respectively. Intuitively, MGC blocks recalibrate representation from shallower stages using deeper semantics so that relevant features are enhanced. This process can be visualized by plotting feature maps before and after they are fed into a MGC block. As shown in Figure 7.12, feature maps from different scales are fed into MGC blocks and enhanced to focus more on UPI-related regions.

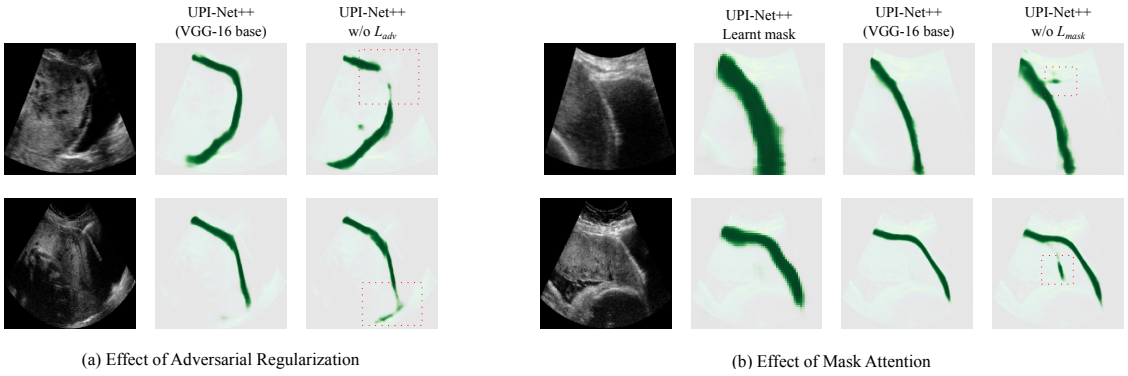


Figure 7.13: Effects of L_{adv} and L_{mask} on UPI detections. (a) L_{adv} enforces spatial continuity. Red dotted boxes highlight some failure cases when the model is trained without L_{adv}. (b) L_{mask} promotes gradient-sensitive spatial attention, which helps filter out unwanted false positives, as marked by the red dotted boxes.

Table 7.4: Effects of L_{adv} in terms of ODS and OIS on validation splits. Two base networks of UPI-Net++, VGG-16 and DRN-D-38, are tested. All results are in the format of *median [first, third quartile]* over six folds.

Base Network	MGC	L_{adv}	L_{mask}	ODS$\uparrow_1$	OIS$\uparrow_1$
VGG-16	✓	✗	✓	0.570 [0.553, 0.586]	0.612 [0.592, 0.623]
VGG-16	✓	✓	✓	0.575 [0.563, 0.595]	0.615 [0.594, 0.631]
DRN-D-38	✓	✗	✓	0.582 [0.563, 0.590]	0.622 [0.599, 0.631]
DRN-D-38	✓	✓	✓	0.596 [0.570, 0.602]	0.631 [0.604, 0.644]

Effect of Adversarial Regularization

One of the auxiliary tasks is to apply adversarial learning on UPI-Net++ to enforce spatial continuity and crispness on its output. Table 7.4 presents experiments to evaluate the effect of L_{adv} on UPI detection by removing it. Quantitatively, this leads to worse performance, as reflected by the reductions of ODS and OIS for both VGG-16 and DRN-D-38. Furthermore, this causes the network to produce more spatially discontinuous UPI predictions, as highlighted by the red dotted boxes in Figure 7.13a. The UPI objective L_{upi} and the adversarial objective L_{adv} in Equation 7.12 complement each other in the sense that minimizing L_{upi} leads to more accurate *pixel-wise* detection whilst L_{adv}, although not explicitly associated with UPI accuracy, regularizes the network's behaviour from a global perspective.

Table 7.5: Effects of L_{mask} in terms of ODS and OIS on validation splits. Two base networks of UPI-Net++, VGG-16 and DRN-D-38, are tested. All results are in the format of *median [first, third quartile]* over six folds.

Base Network	MGC	L_{adv}	L_{mask}	ODS$\uparrow_1$	OIS$\uparrow_1$
VGG-16	✓	✓	✗	0.568 [0.558, 0.583]	0.604 [0.598, 0.621]
VGG-16	✓	✓	✓	0.575 [0.563, 0.595]	0.615 [0.594, 0.631]
DRN-D-38	✓	✓	✗	0.583 [0.564, 0.599]	0.618 [0.603, 0.641]
DRN-D-38	✓	✓	✓	0.596 [0.570, 0.602]	0.631 [0.604, 0.644]

Table 7.6: Performance of UPI detectors on the test splits. All results are in the format of *median [first, third quartile]* over six folds.

Models	Base Network	ODS$\uparrow_1$	OIS$\uparrow_1$
HED [140]	VGG-16	0.528 [0.521, 0.576]	0.576 [0.571 0.621]
CASENet [174]	VGG-16	0.550 [0.541, 0.588]	0.589 [0.580, 0.625]
DS-FPN [116]	VGG-16	0.549 [0.530, 0.583]	0.584 [0.569, 0.618]
STEAL [176]	VGG-16	0.549 [0.539, 0.577]	0.589 [0.577, 0.614]
UPI-Net § 6.2.2	VGG-16	0.555 [0.532, 0.570]	0.587 [0.567, 0.601]
UPI-Net++	VGG-16	**0.560 [0.546, 0.600]**	0.594 [0.589, 0.643]
HED [140]	DRN-D-38	0.558 [0.534, 0.571]	0.597 [0.572, 0.612]
CASENet [174]	DRN-D-38	0.556 [0.546, 0.570]	0.588 [0.578, 0.604]
DS-FPN [116]	DRN-D-38	0.556 [0.546, 0.579]	0.587 [0.574, 0.609]
STEAL [176]	DRN-D-38	0.559 [0.545, 0.583]	0.590 [0.581, 0.616]
UPI-Net § 6.2.2	DRN-D-38	0.563 [0.551, 0.595]	0.590 [0.583, 0.631]
UPI-Net++	DRN-D-38	**0.572 [0.562, 0.630]**	0.608 [0.604, 0.665]

Effect of Mask Attention

The mask branch of UPI-Net++ predicts a spatial attention map that helps filter out unwanted false positives that do not fall inside the UPI neighbourhood, as shown in Figure 7.13b. Experimental results in Table 7.5 further indicate that this auxiliary mask segmentation task improves UPI detection performance. More examples are demonstrated in Figure 7.14. It appears that UPI-Net++ learnt to predict masks that are also gradient-sensitive.

Comparison to Benchmark Models

The following competitive benchmark models were implemented as comparisons to the proposed UPI-Net++ : Holistically-nested edge detector (HED) [140]; CASENet

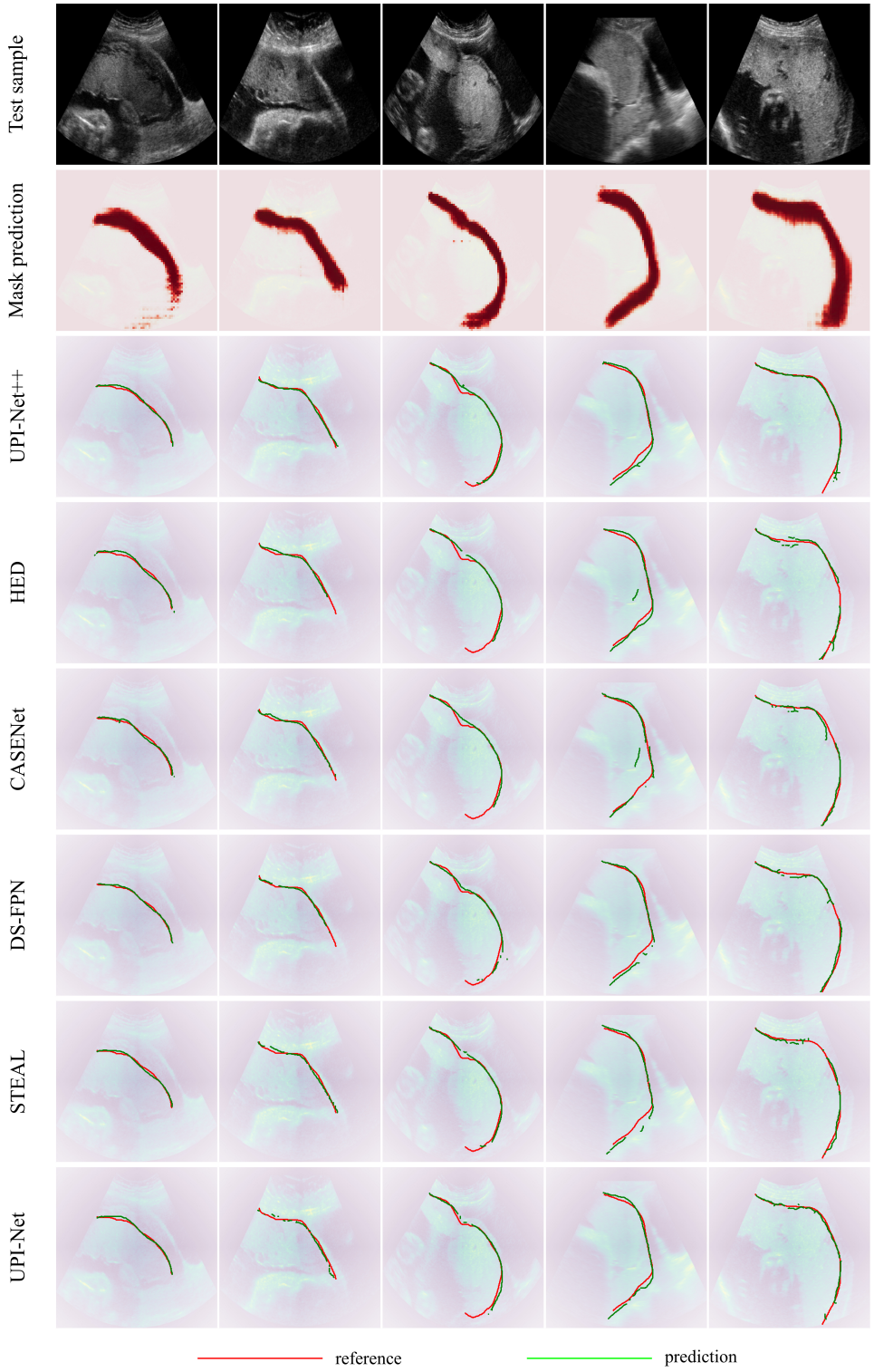


Figure 7.14: Predictions from the proposed UPI-Net++ and other benchmark models. As with Figure 6.10, red curves denote the reference UPI and green curves denote UPI predictions.

[174]; DS-FPN [116]; STEAL[1] [176]; UPI-Net (§ 6.2.2). Note that stride-2 max-pooling layers in the last three stages of these networks are replaced with dilated convolutions to maintain spatial resolution. Table 7.6 shows that the proposed UPI-Net++ outperforms all other competitive benchmark models in terms of ODS and OIS. Two-tailed paired t-tests show that ODS and OIS scores of UPI-Net++ are significantly higher than those from the rest benchmark models (with p-value < 0.001). In general, DRN-D-38 gives a better performance than VGG-16 as the base network thanks to the residual structure throughout the model. Some predictions from test splits are shown in Figure 7.14. UPI-Net++ performs better than the rest models in terms of suppressing unwanted false positive predictions and also maintaining spatial continuity.

7.3.4 Limitations

UPI-Net++ exploits adversarial learning to regularize network behaviours. However, this leads to one significant limitation: the adversarial network introduces notable computational overhead during training. Moreover, instability caused by adversarial learning [77] can not be overlooked. A future direction would be to explore more effective and efficient approach such as light-weighted architectural refinement or a new form of loss function that achieves similar effects as adversarial learning does.

7.4 Conclusions

This chapter has presented a novel approach to UPI detection by solving two auxiliary tasks along with the primary task. The first auxiliary task is to predict a spatial attention mask that explicitly attends the UPI-related regions via a mask attention module. Without any extra annotation effort, the associated supervision signal comes from the reference UPI mask through a gradient-sensitive transformation. It was found that the learnt attention mask can effectively filter out unwanted false positives that lie far away from the UPI neighbourhood. The second task is

[1]The non-maximum suppression loss was implemented via bilinear sampling [212] while the direction loss was removed due to the unstable gradient flow it caused

to further regularize network's behaviours by introducing an adversarial objective function. It intends to enforce spatial continuity and crispness of network outputs by training an extra classification network to distinguish Gaussian density maps from UPI predictions. Similarly, the Gaussian density maps are generated directly from the reference UPI masks using distance transform without any additional cost. A new multi-scale feature aggregation module named MGC block is proposed to guide representation learning in UPI-Net++. It outperforms the GC block used in § 6.2.2 by passing high-level semantics from deeper stages to recalibrate low-level features. Experimental results demonstrate the superiority of the proposed UPI-Net++ in terms of ODS and OIS.

8

Image-based Prenatal Assessment of Placenta Accreta Spectrum Disorders

Contents

8.1 Background and Motivation 145
8.2 Image-based Prenatal Assessment Pipeline 148
8.2.1 Dataset and Training Protocol 148
8.2.2 Informative Plane Screening 149
8.2.3 Plane-wise Pattern Recognition 155
8.2.4 Collaborative PASD Assessment 156
8.2.5 Experimental Results 159
8.3 Comparison to Human Experts 161
8.4 Conclusions 167

This chapter presents an automated prenatal screening pipeline for PASD by integrating a sequence of computational modules that accomplish specific tasks. Two key pattern recognition modules are from Chapter 5 and Chapter 7. The pipeline takes a 3D placental ultrasound volume as an input and predicts its PASD label through a series of placental image analysis operations. To evaluate the performance of the automated screening pipeline against human experts, a panel of PASD researchers and clinicians with years of relevant experience were invited to estimate PASD labels given placental ultrasound volumes alone. A variety of statistical analysis were carried out to validate the proposed machine-learning-based

method. Experimental results suggest that the proposed screening pipeline has the potential to assist clinicians in the prenatal diagnosis of PASD.

8.1 Background and Motivation

During the past decade, emerging machine learning paradigms (e.g. deep learning) as well as increasing computational power and accumulation of 'big data' have advanced a few key areas of medicine by achieving physician-level performance in various tasks [213–219]. In particular, studies in medical imaging have demonstrated promising results in diagnostics covering pathology [215, 220, 221], radiology [218, 222–224], ophthalmology [216, 217], etc. For instance, Esteva et al. developed a deep neural network model that identified skin cancers given images of skin lesions. The model was trained on more than 100,000 clinical images in an end-to-end fashion and achieved dermatologist-level classification performance [214]. De Fauw et al. applied a deep learning architecture to a dataset of around 14,000 3D optical coherence tomography scans to build up a clinically applicable referral system that reached expert-level performance on a range of sight-threatening retinal diseases [216]. Their approach consisted of a 3D segmentation network to perform voxel-level classification and a classification network that mapped tissue-segmentation maps to diagnostic decisions and referral suggestions. Coudray et al. developed a tile-based deep-learning model to assess the type and sub-type of lung tumors by examining lung histopathology images containing nearly 1,000,000 tiles. The performance was comparable to that of pathologists [215]. Their findings suggested that deep-learning models have the potential to assist pathologists in the detection of cancer subtype or gene mutations. Overall, these breakthroughs demonstrate that machine-learning-based systems could improve diagnostic efficacy, streamline clinician workflow and provide reliable second opinions.

Traditionally, domain expertise and feature engineering are required to construct a machine-learning system that maps raw data into suitable representations via carefully designed algorithmic pipeline. By contrast, deep-learning approaches rely on data-driven representation learning to develop complex feature extractors needed

for pattern recognition via multi-layer non-linearity sequences [52]. To make full use of this capability, the aforementioned studies used large sets of well-annotated medical data to train their deep learning models since it would be difficult for deep-learning models to differentiate between relevant and irrelevant features given a small set of high-dimensional data (e.g. images) [225]. This leads to the first significant limitation of applying deep-learning algorithms in medical practice: lack of training data. For many medical applications, it is not applicable to build up datasets of similar orders of magnitudes as those large-scale studies did. One reason is that collecting and annotating large datasets can be an infeasibly time-consuming and costly process in many aspects. Another possible reason is the relative rarity of some conditions. Low incidence rate makes collecting a large set of data impossible in such conditions. Another limitation is that the decision-making processes of deep-learning models remain non-transparent and lack explainablity [219, 226]. A vast majority of medical imaging studies develop deep-learning models in an end-to-end fashion, i.e. transforming raw image data into desired outputs in one single feed-forward computation without any intermediate feedback for inspection. This 'black-box' behaviour hinders wider adoption of deep learning in clinical diagnostic processes.

Prenatal assessment of PASD aims to categorize an input placental ultrasound scan into one of the three PASD classes: normal placenta (NP), abnormally adherent placenta (AAP), or abnormally invasive placenta (AIP) (see § 3.2). This chapter attempts to overcome limitations of deep-learning methods by developing a prenatal screening pipeline for PASD, using a small and imbalanced 3D placental ultrasound dataset, i.e. the Oxford-Hackensack database of PASD. Instead of training a single end-to-end 3-way classification network, the proposed pipeline breaks down the task of PASD assessment into a sequence of sub-tasks, each approached by a specific computational module. As shown in Figure 8.1, there are four computational modules wrapped in three key stages:

- Stage 1: Informative Plane Screening. A key pre-processing step in the proposed pipeline would be to determine whether an image plane is informative for PASD assessment. This stage takes a 3D placental ultrasound volume

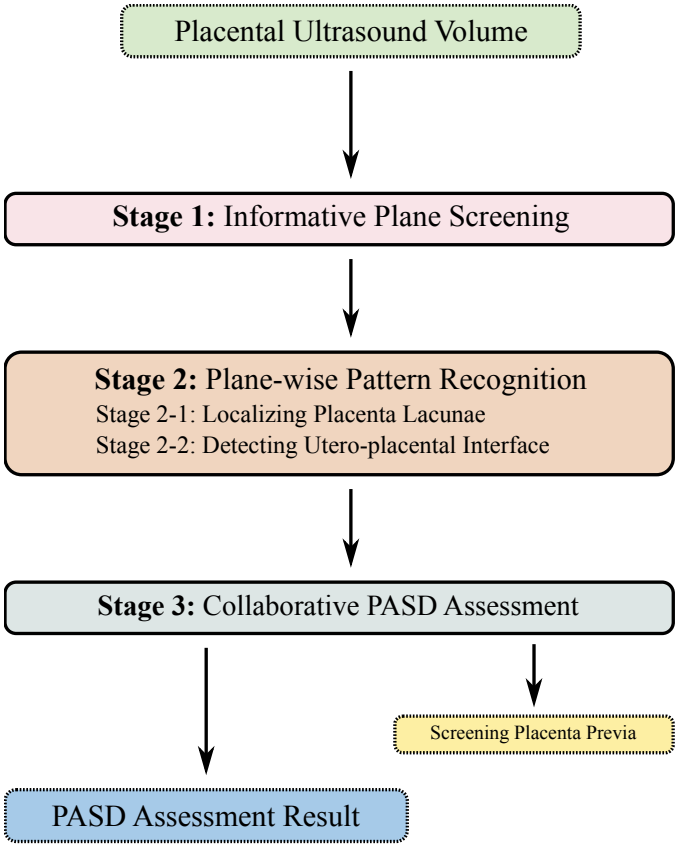


Figure 8.1: An image-based prenatal assessment pipeline for PASD, with three key stages enclosing four computational modules.

scan as an input and distinguishes informative sagittal planes from the non-informative ones by training a deep-learning-based *informative plane screening network*. This stage conducts important data pruning in order to exclude noisy or troublesome planes from the dataset.

- Stage 2: Plane-wise Pattern Recognition. Important ultrasound signs of PASD are related to placental lacunae (PL) and utero-placental interface (UPI). This stage carries out deep-learning-based pattern recognition to localize PL and detect the UPI in a plane-wise manner. The resulting predictions highlight the clinically relevant regions in placental ultrasound images for PASD assessment.

- Stage 3: Collaborative PASD Assessment. Having collected necessary anatomical patterns, this stage deploys a deep-learning-based *PASD assessment network* (PAN) that integrates informative planes, estimated placental lacunae

heatmaps and UPI masks in a collaborative way to perform PASD classification (primary task) and an additional placental previa classification (auxiliary task). The resulting plane-wise predictions are further aggregated to form a volume-wise prediction.

Unlike an end-to-end 'black-box' model, the proposed three-stage screening pipeline breaks down the comprehensive task of PASD assessment into a sequence of clinically relevant tasks and produces intermediate results to enhance explainability. In some sense, it mimics the workflow of a clinician who attends to regions of interest, locates the abnormality and performs a diagnosis by combining feedback from different perspectives.

8.2 Image-based Prenatal Assessment Pipeline

This section demonstrates the workflow of the proposed image-based prenatal assessment pipeline in a stage-by-stage manner. First, dataset statistics and the overall training/testing protocol are summarized in § 8.2.1. The next three subsections present architectures, implementation details and experimental results of the three key stages: informative plane screening (§ 8.2.2), plane-wise pattern recognition (§ 8.2.3), collaborative PASD assessment (§ 8.2.4).

8.2.1 Dataset and Training Protocol

As stated in Chapter 3, the Oxford-Hackensack database of PASD contains 101 3D placental ultrasound volumes with histopathologically confirmed PASD screen labels and previa screen labels (see § 3.2). The label distributions and the training/testing protocol are illustrated in Figure 8.2. There are 53 NP volumes, 18 AAP volumes and 30 AIP volumes (Figure 8.2a), or 5,759 NP planes, 1,978 AAP planes and 3,429 AIP planes. The exact same 6-fold nested cross-validation protocol from Chapter 7 is adopted in this chapter. As presented in Table 8.1, each fold has a held-out test split containing 16-17 volumes and the number of NP/AAP/AIP cases are distributed according to their overall frequencies in the database. The rest volumes are divided

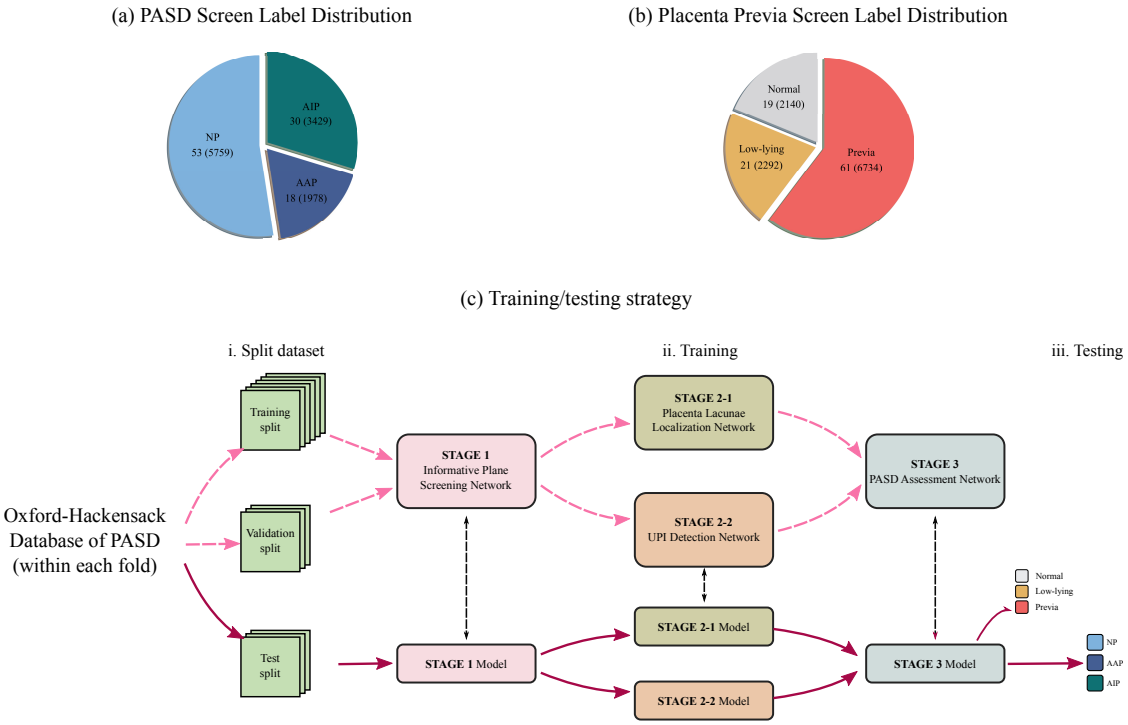


Figure 8.2: Data distribution and training/test protocol. (a) PASD screen label distribution with respect to placental ultrasound volumes (planes); (b) placenta previa screen label distribution with respect to placental ultrasound volumes (planes); (c-i) the dataset is first divided into training splits, validation splits and test splits; (c-ii) deep-learning models are trained on training splits and tuned on validation splits; (c-iii) trained models from each stage are then evaluated on test splits. As with § 7.3.1, the same 6-fold nested cross-validation are adopted to compensate for the lack of test data.

into a training split and a validation split (Figure 8.2c-i). Samples from the training and validation splits are used to train computational modules whilst evaluations are performed solely on the held-out test splits (Figure 8.2c-ii,iii). This protocol guarantees that models are never trained and tested on image planes obtained from the same placental ultrasound volume. These operations are conducted repeatedly on six non-overlapping folds in order to compensate for the lack of test data.

8.2.2 Informative Plane Screening

In the first stage of the proposed PASD assessment pipeline, an *informative plane screening network* (ResNet-18 [84]) is developed that classifies placental ultrasound image planes into *informative* and *non-informative* planes. The supervision signals are the image information labels (see § 3.2.2). Among 58,202 image planes in the

Table 8.1: Test splits of a 6-fold nested cross-validation protocol. Each fold contains 16 or 17 non-overlapping placental ultrasound volumes, which takes up 101 volumes in total. Volumes belonging to NP/AAP/AIP categories are randomly distributed to six folds according to their frequencies.

Test splits	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Total
Volumes	17	17	17	16	17	17	101
NP	10	7	9	9	8	10	53
AAP	4	4	3	2	3	2	18
AIP	3	6	5	5	6	5	30

database, 11,166 were marked as being informative. To handle this imbalanced dataset, a weighted binary cross-entropy objective function is applied to train the network, with the weight associated with *informative* class being set to the frequency of *non-informative* samples in the training set and vice versa. All placental ultrasound images in the database are rescaled to $[0, 1]$, zero-centred by subtracting the mean and resized to 320×320 pixels. A series of data augmentation operations are conducted:

- Each side of the image is padded or cropped by a randomly sampled ratio between 0 and 10% relative to its original size and resized to 320×320 pixels.
- Each patch is translated along x-axis and y-axis by a randomly sampled ratio between 0 and 10% relative to its original size and resized to 320×320 pixels.
- Gamma correction and logarithm correction are performed following Equation 6.6.

The hyper-parameter configuration is as follows. The mini-batch size is set to 32. An AdamW optimizer is adopted with the initial learning rate of 0.001 [74, 75]. The weight decay is set to 0.0005. There are two options for ResNet-18 initialization. The first one is to initialize it with the ImageNet pretrained parameters and conduct transfer learning by fine-tuning for the task of informative plane classification. The second one is to completely train from scratch by initializing the network with zero-mean Gaussian distributions, following the method in [177]. Both strategies were tested and reported. The models were implemented with PyTorch and trained

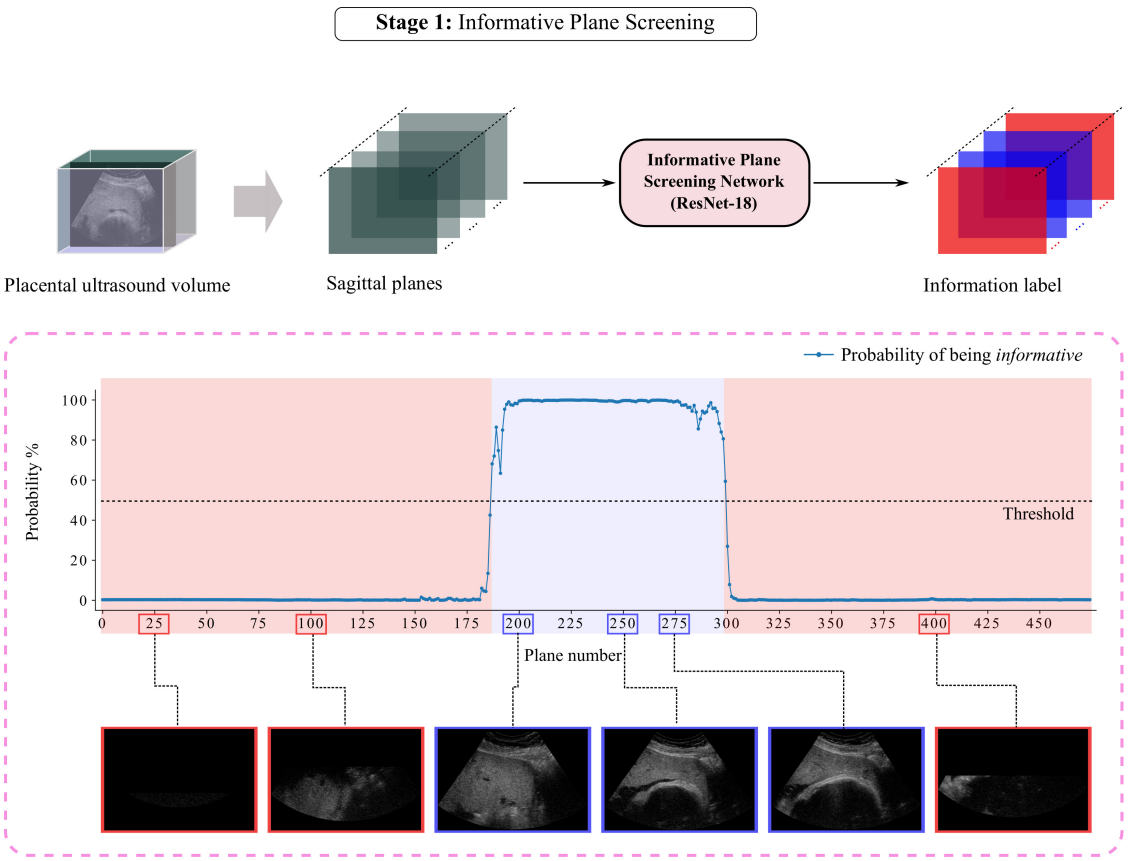


Figure 8.3: Stage 1 workflow. Raw placental ultrasound volumes are sliced along sagittal planes and classified by the informative plane screening network. An illustrative sample is presented inside the dotted box, where the plane-wise probabilities are plotted by varying plane indices.

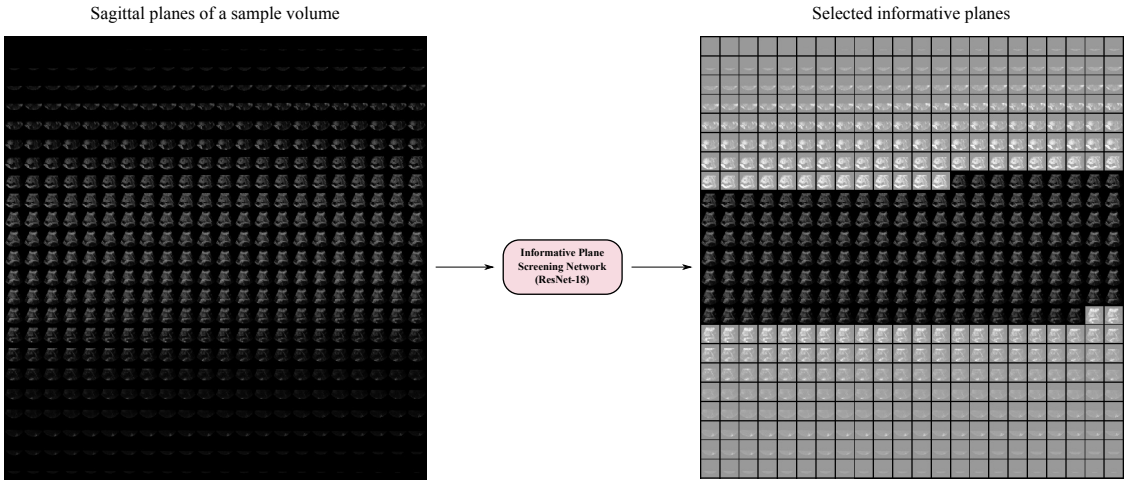


Figure 8.4: An illustrative example of applying the *informative plane screening network* on the sagittal planes of a sample volume. In the right figure, planes with white backgrounds are filtered out by the model and the remaining ones are selected for subsequent analysis.

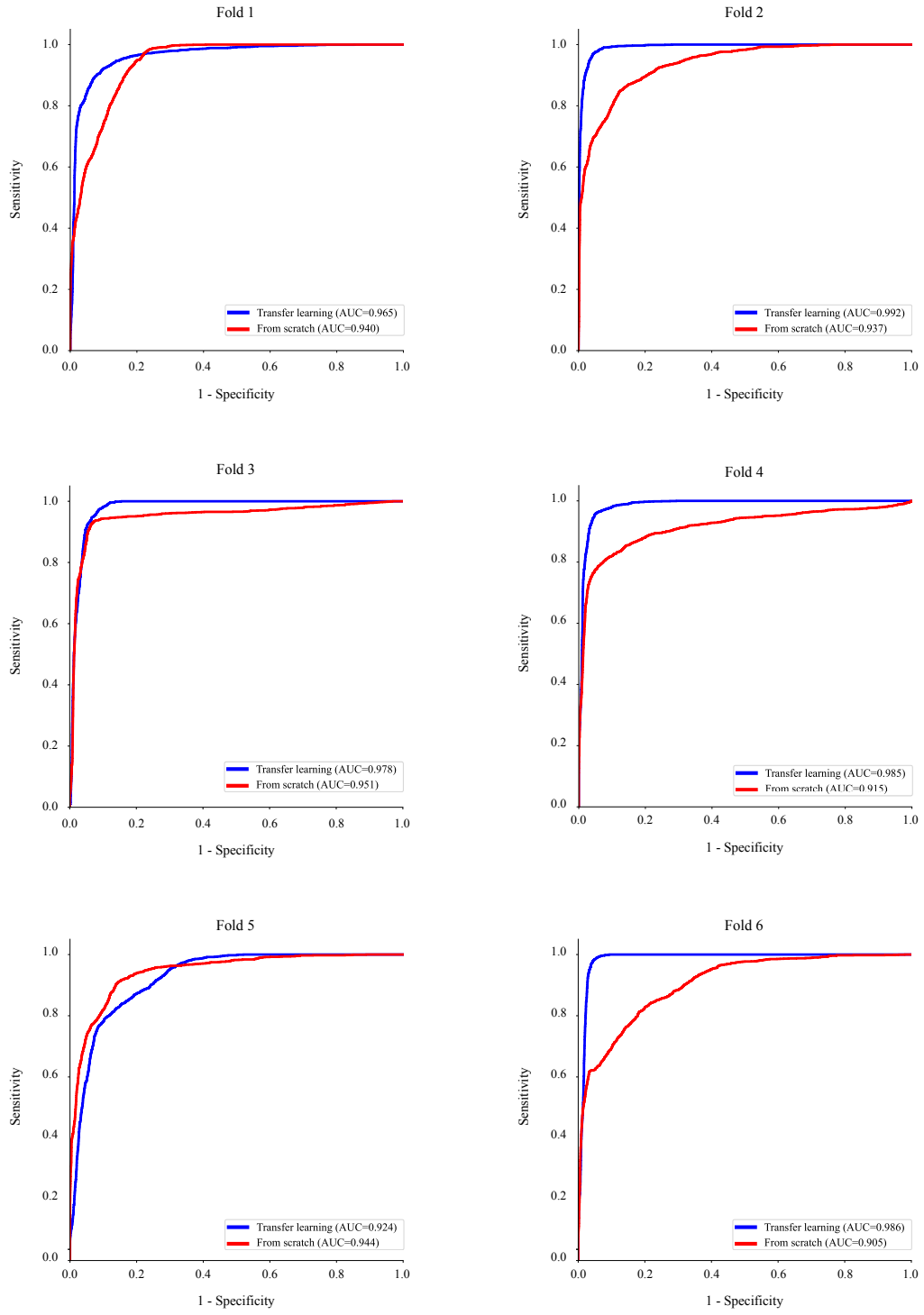


Figure 8.5: ROC curves for informative plane screening networks. The blue curves denote models trained via transfer learning while the red curves denote those trained from scratch.

for 40 epochs with early stopping (patience level set to 7 epochs) on an NVIDIA DGX-1 with P100 GPUs. The entire workflow of this stage is summarized in

Table 8.2: Model performance for informative plane screening on test splits via transfer learning from ImageNet-pretrained ResNet-18. The 95% confidence intervals (CIs) on the metrics are written in parentheses.

Transfer learning	AUC	Sensitivity (%)	Specificity (%)
Fold 1	0.965 (0.960, 0.969)	95.8 (95.3, 96.2)	81.6 (80.8, 82.4)
Fold 2	0.992 (0.991, 0.993)	98.4 (98.2, 98.6)	93.8 (93.3, 94.3)
Fold 3	0.978 (0.975, 0.980)	99.6 (99.5, 99.7)	88.0 (87.4, 88.6)
Fold 4	0.985 (0.983, 0.987)	92.2 (91.7, 92.7)	96.5 (96.1, 96.9)
Fold 5	0.924 (0.918, 0.930)	94.9 (94.5, 95.3)	70.5 (69.6, 71.4)
Fold 6	0.986 (0.983, 0.988)	98.4 (98.1, 98.7)	95.3 (94.9, 95.7)

Table 8.3: Model performance for informative plane screening on test splits via random Gaussian parameter initialization following [177]. The 95% confidence intervals (CIs) on the metrics are written in parentheses.

From scratch	AUC	Sensitivity (%)	Specificity (%)
Fold 1	0.940 (0.936, 0.945)	98.8 (98.6, 99.0)	75.3 (74.4, 76.2)
Fold 2	0.937 (0.931, 0.942)	96.0 (95.6, 96.4)	65.3 (64.4, 66.2)
Fold 3	0.951 (0.944, 0.958)	93.6 (93.1, 94.1)	92.3 (91.8, 92.8)
Fold 4	0.915 (0.905, 0.924)	89.3 (88.7, 89.9)	77.6 (76.8, 78.4)
Fold 5	0.944 (0.939, 0.949)	94.2 (93.7, 94.7)	79.6 (78.8, 80.4)
Fold 6	0.905 (0.898, 0.912)	93.5 (93.0, 94.0)	63.8 (62.8, 64.8)

Figure 8.3. Figure 8.4 presents screening results on a sample volume.

Fold-wise receiver operating characteristic (ROC) curves are displayed in Figure 8.5, along with the area under ROC curve (AUC) metric, for models finetined on ImageNet-pretrained ResNet-18 as well as those trained from scratch. Overall, the proposed informative plane screening networks achieve AUCs larger than 0.9 on all test splits. It appears that transfer learning models have superior classification performance over their counterparts that are trained from scratch. Furthermore, Table 8.2 and Table 8.3 show the sensitivity and specificity measures of each fold, which are computed by using the threshold of high sensitivity operating points from the validation splits since it is clinically motivated to maximize sensitivity for informative plane screening. Note that the 95% confidence intervals (CIs) are also presented following previous studies [214–216]. A non-parametric bootstrap method is used to calculate the 95% CIs for AUCs. Specificially, for a test split of size n, n samples were randomly sampled with replacement to estimate the corresponding AUC. This process was repeated for 1,000 times and 95% CIs

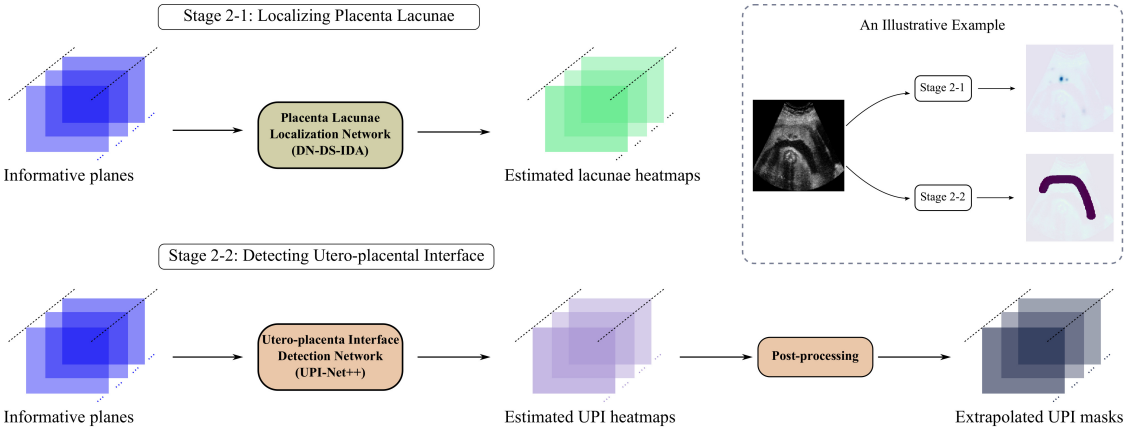


Figure 8.6: Stage 2 workflow. Informative planes are fed into two pattern recognition branches for accurate PL localization and UPI detection respectively.

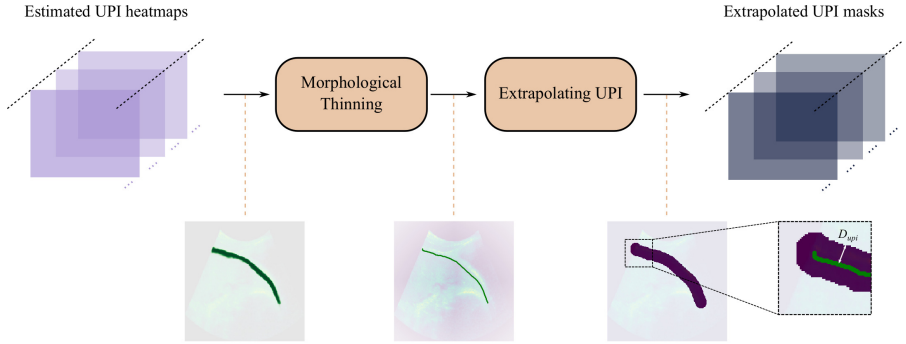


Figure 8.7: UPI post-processing module in Stage 2-2. It takes UPI heatmaps from UPI-Net++ as inputs and performs anatomically grounded extrapolation to generate UPI masks that serve as spatial attention maps for subsequent analysis.

were represented as the 2.5 and 97.5 percentiles from the AUC sequence. The 95% CIs of sensitivity and specificity were computed using binomial proportion CIs $= [\mathrm{CI}_{low}, \mathrm{CI}_{high}]$[1], which are obtained by:

$$\mathrm{CI}_{low} = p - z\sqrt{\frac{p(1-p)}{n}} \quad \mathrm{CI}_{high} = p + z\sqrt{\frac{p(1-p)}{n}} \tag{8.1}$$

where p is the measure of interest, n is the sample size and z is the $(1 - \frac{\alpha}{2})$ quantile of a standard Gaussian distribution for an error rate α. For 95% CIs, the error $\alpha = 0.05$ and $z \approx 1.96$.

[1] https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval

Table 8.4: Performance (%) of the DN-DS-IDA model on test splits.

DN-DS-IDA	mAP	$AP_{0.3}$	$AP_{0.5}$	$AP_{0.7}$	$AP_{0.9}$
Fold 1	49.0	65.5	60.4	51.2	21.4
Fold 2	49.6	64.4	60.8	51.3	24.1
Fold 3	40.4	53.5	49.3	41.1	20.7
Fold 4	41.0	52.5	49.5	43.4	21.0
Fold 5	47.8	62.2	58.9	49.3	21.3
Fold 6	47.1	63.5	58.6	48.2	18.5

8.2.3 Plane-wise Pattern Recognition

The second stage of the proposed PASD assessment pipeline focuses on extracting key anatomical patterns from informative planes, namely placental lacunae (PL) and utero-placental interface (UPI). Specifically, a *placenta lacunae localization network* (DN-DS-IDA, § 5.3) and a *utero-placental interface detection network* (UPI-Net++, § 7.2.3) are deployed in two branches of this stage. Figure 8.6 illustrates the workflow during testing, where an input image plane is fed to two modules respectively to obtain corresponding PL and UPI prediction heatmaps. Since almost all technical details of these two networks have been covered in Chapter 5, Chapter 6 and Chapter 7, this chapter only complements them with some design choices and additional experimental results:

1. Both DN-DS-IDA and UPI-Net++ use the ImageNet-pretrained dilated residual network DRN-D-38 as their base networks [178] as it gives superior performance on validation splits in both tasks compared to the rest base networks.

2. In Chapter 5, DN-DS-IDA was only trained on a subset of the Oxford-Hackensack database of PASD (see § 3.3 for the database's relation to each chapter). Therefore, the experimental results in Chapter 5 is not well-suited for reference. Following the training protocol in § 8.2.1, a DN-DS-IDA model is re-trained on the whole dataset[2]. Table 8.4 presents the average precision (AP) metrics of this model on test splits.

[2]The reason for adopting DN-DS-IDA rather than UN-DS-IDA is to reduce memory footprint as U-shape networks (UN) require larger memory allocation compared with its down-sampling network (DN) counterpart.

3. Stage 2-2 contains a UPI post-processing module. As shown in Figure 8.7, estimated UPI heatmaps from UPI-Net++ are first thinned to 1-px wide via morphological operation [200]. Anchored on the thinned UPI curve, an extrapolation operation is then conducted via distance transform to obtain a UPI mask enclosing all spatial locations that are within a distance of D_{upi}. The resulting mask serves as a spatial attention map that highlights a key region of interest, which an experienced sonographer would normally examine during PASD assessment. The meaning of D_{upi} is anatomically grounded since all the placental ultrasound volumes in the database are spatially sampled at the rate of 0.33 mm/voxel. As shown in the bottom-right of Figure 8.7, D_{upi} essentially parameterizes how far away from UPI the attention should be expanded for PASD assessment. This chapter performs PASD assessment by setting D_{upi} at $3mm$, $6mm$ and $9mm$ (or 9 pixels, 18 pixels and 27 pixels) respectively, which allows investigations on how the size of spatial attention maps centered at UPI affects assessment performance.

8.2.4 Collaborative PASD Assessment

The third stage of the proposed PASD assessment pipeline introduces a *PASD assessment network* (PAN) that integrates knowledge from previous stages in a collaborative way for plane-wise PASD classification and placenta previa classification. Volume-wise classification scores are obtained by aggregating plane-wise scores. The workflow of Stage 3 is illustrated in Figure 8.8.

The intuition behind Stage 3 is straight-forward. In order to achieve good classification performance on a relatively small dataset, it is important for the model to focus on anatomies that are clinically relevant in order to guide representation learning. As stated in § 1.2.2, most sonographic patterns associated with PASD assessment are spatially close to UPI. This justifies the use of a UPI mask (from Stage 2-2) that provides spatial attention to the model, i.e. explicitly guiding the model to learn discriminative feature representation in the neighbourhood of UPI. Another important ultrasound sign related to PASD is the presence of numerous

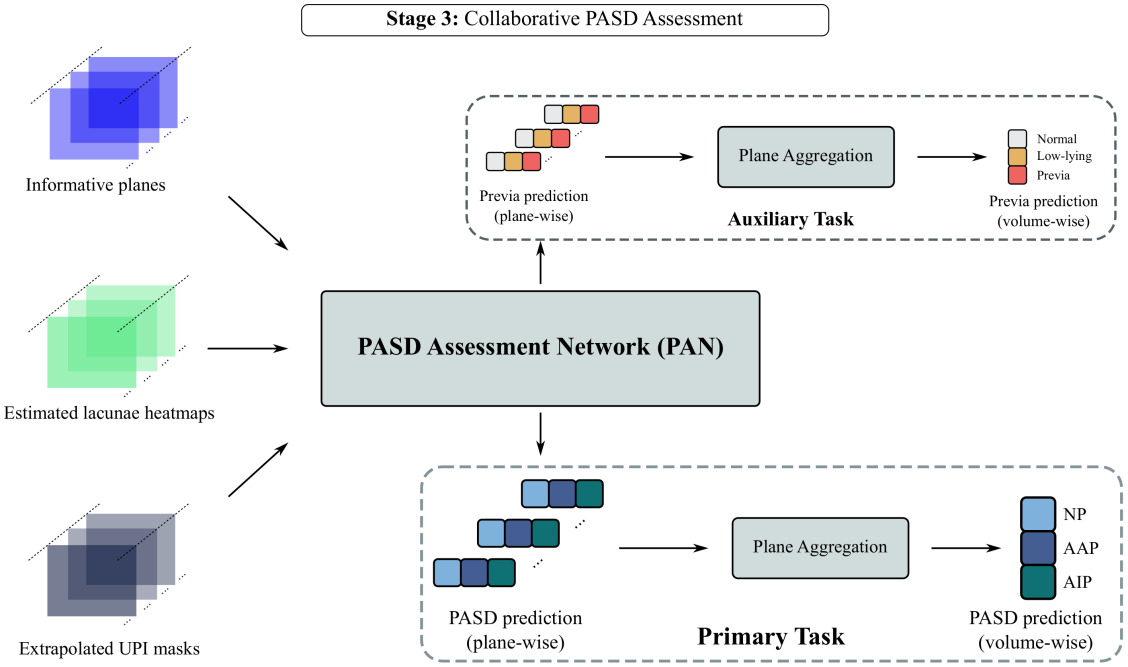


Figure 8.8: Stage 3 workflow. The proposed PAN takes as input a tuple of information: an informative plane, its estimated PL heatmap from Stage 2-1 and its extrapolated UPI mask from Stage 2-2. It then outputs PASD classification scores (primary task) and placenta previa classification scores (auxiliary task) in a plane-wise manner. Volume-wise scores are calculated via plane aggregation.

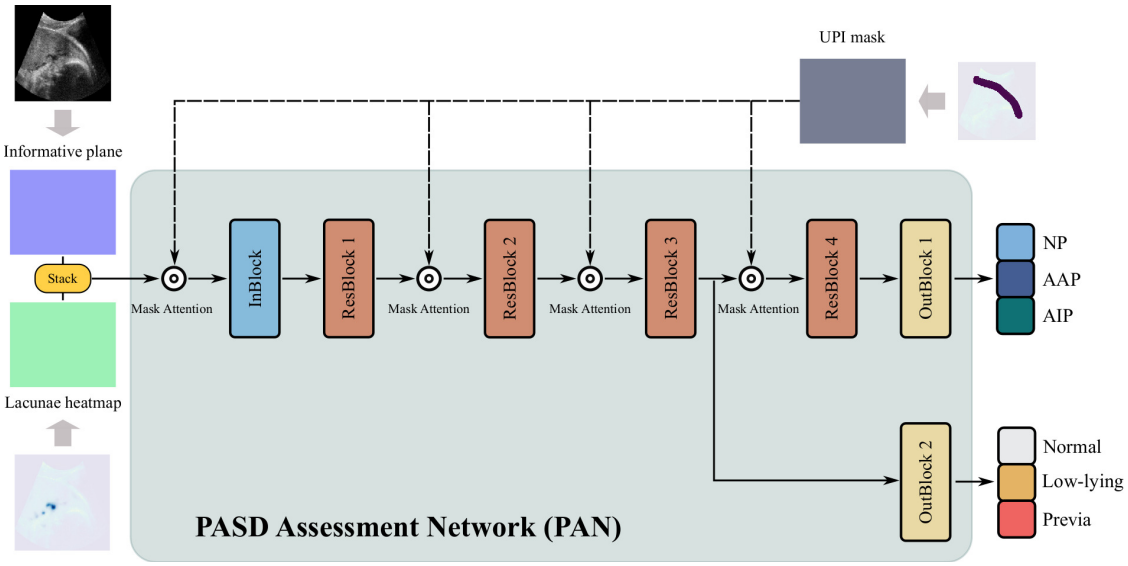
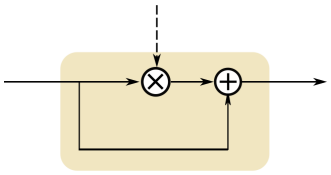


Figure 8.9: Architecture of PAN.

lacunae (see Table 1.2), which supports the use of a PL heatmap (from Stage 2-1). Working collaboratively, these anatomical patterns are expected to facilitate representation learning thus lead to better classification performance compared



(a) Mask Attention Block

	InBlock	ResBlock 1	ResBlock 2
Structure	7×7 64 stride 2	$\begin{bmatrix} 3\times3 & 64 \\ 3\times3 & 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3 & 128 \\ 3\times3 & 128 \end{bmatrix} \times 2$
Output size	160×160	160×160	80×80
	ResBlock 3	ResBlock 4	OutBlock 1&2
Structure	$\begin{bmatrix} 3\times3 & 256 \\ 3\times3 & 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3 & 512 \\ 3\times3 & 512 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{average pooling} \\ \text{dropout}(0.5) \\ \text{fc}(512 \rightarrow 3) \end{bmatrix}$
Output size	40×40	20×20	1×1

(b) PAN Architecture (based on ResNet-18)

Figure 8.10: (a) A mask attention block to enforce UPI spatial attention; (b) Convolutional layers inside PAN's blocks.

to models trained solely on informative planes without collaborative mechanism. To this end, this chapter proposes PAN, a deep convolutional neural network that conducts collaborative PASD assessment based on residual blocks and mask attention blocks. It is based on a ResNet-18 model [84]. Figure 8.9 illustrates PAN's architecture. First, an image plane and its PL heatmap are stacked along the channel axis[3] to form an input tensor while the UPI mask serves as a spatial attention map to guide feature learning via mask attention blocks (Figure 8.10a). The mask attention block adopts a residual pathway to ensure that while the UPI neighbourhood is a key region of interest, global context is retained in PAN. Four mask attention blocks are placed in PAN: at the beginning, after ResBlock 1, after ResBlock 2 and after ResBlock 3 to enforce UPI spatial attention across multiple scales. PAN has two classification heads, targeting PASD classification and placenta previa classification respectively. Reference PL confidence maps and extrapolated UPI reference binary masks are used during training, whilst the estimated PL heatmaps and extrapolated UPI masks from Stage-2 are used during testing since reference annotations should not be accessed except for performance evaluation. The overall objective function is given by:

$$L_{all} = L_{PASD} + \lambda_p L_{previa} \tag{8.2}$$

where both L_{PASD} and L_{previa} are weighted cross-entropy functions and λ_p is a weighting parameter to control the importance of the auxiliary task. It is empirically set to 0.5.

[3]Note that ImageNet-pretrained ResNet-18 takes a 3-channel tensor, hence the input image plane duplicates itself and stacks with its PL heatmap to form a 3-channel tensor.

By design, PAN carries out plane-wise assessment. In order to perform volume-wise evaluation, the per-plane classification results were aggregated on a per-volume basis by averaging the probabilities obtained for each plane. This is referred to as plane aggregation. PAN adopts the same data augmentation operations as the informative plane screening network in § 8.2.2. The hyper-parameter configuration is as follows. The mini-batch size is set to 6. The initial learning rate of the AdamW optimizer is set to 0.001. The weight decay is set to 0.05. Transfer learning is conducted based on an ImageNet-pretrained ResNet-18 model. PAN was implemented with PyTorch and trained for 40 epochs with early stopping patience level of 7 epochs on an NVIDIA DGX-1 with P100 GPUs.

8.2.5 Experimental Results

PASD assessment performance of PAN is evaluated on six non-overlapping test splits, following the 6-fold nested cross-validation protocol in § 8.2.1. According to § 8.2.3, three extrapolation distances are considered: D_{upi} = 3 mm, 6 mm, 9 mm. This corresponds to the spatial extent of UPI neighbourhood that is considered relevant for PASD assessment. Experimental results are shown in Table 8.5. Class-wise AUCs are measured by considering each class as a binary problem, i.e. learning to predict each class against the rest. Both plane-wise AUC and volume-wise AUC are reported, corresponding to classification results before and after plane aggregation. To compare against the naive end-to-end image classification approach, a baseline ResNet-18 model was trained using the same protocol and hyper-parameter configuration as PAN. From now on, PAN trained with D_{upi} = 3 mm, 6 mm, 9 are referred to as *PAN-3, PAN-6 and PAN-9* while the baseline ResNet-18 is referred to as *the baseline* for simplicity.

For better illustration, entries with the highest class AUC on certain classes across models are marked bold in Table 8.5. It is clear that PAN outperforms the baseline in terms of volume-wise AUC across test splits and PASD classes. Moreover, PAN-6 and PAN-9 perform slightly better than PAN-3, which can be attributed to the larger UPI spatial attention. Chapter 4 introduced class

Table 8.5: Volume-wise and plane-wise AUC (with 95% CIs) achieved by PAN-3, PAN-6, PAN-9 and the baseline ResNet-18 for each PASD class respectively. Entries with highest AUC on each class across models are marked as **bold**.

NP	PAN-3		PAN-6		PAN-9		ResNet-18	
	Volume	Plane	Volume	Plane	Volume	Plane	Volume	Plane
Fold 1	0.857	0.781 (0.760,0.803)	**0.943**	0.868 (0.851,0.884)	0.914	0.845 (0.827,0.861)	0.757	0.732 (0.709, 0.755)
Fold 2	0.857	0.810 (0.791,0.828)	0.857	0.803 (0.783,0.823)	**0.929**	0.872 (0.855,0.888)	0.771	0.729 (0.706,0.751)
Fold 3	**0.750**	0.709 (0.684,0.734)	0.708	0.727 (0.701,0.753)	0.736	0.777 (0.755,0.797)	0.611	0.654 (0.628,0.682)
Fold 4	**0.905**	0.883 (0.869,0.897)	0.889	0.829 (0.810,0.848)	0.889	0.858 (0.843,0.873)	0.873	0.819 (0.798,0.838)
Fold 5	0.778	0.678 (0.653,0.704)	**0.833**	0.716 (0.691,0.738)	0.764	0.630 (0.602,0.656)	0.389	0.356 (0.330,0.381)
Fold 6	0.857	0.783 (0.763,0.802)	0.829	0.783 (0.761,0.802)	**0.900**	0.805 (0.785,0.824)	0.671	0.616 (0.592,0.642)
AAP	PAN-3		PAN-6		PAN-9		ResNet-18	
	Volume	Plane	Volume	Plane	Volume	Plane	Volume	Plane
Fold 1	0.596	0.508 (0.477,0.538)	**0.769**	0.698 (0.662,0.731)	0.538	0.490 (0.456,0.524)	0.596	0.615 (0.583,0.647)
Fold 2	0.519	0.486 (0.456,0.516)	**0.750**	0.593 (0.562,0.623)	0.654	0.636 (0.608,0.662)	0.577	0.591 (0.565,0.617)
Fold 3	0.571	0.613 (0.585,0.640)	**0.762**	0.663 (0.632,0.689)	0.738	0.636 (0.600,0.672)	0.738	0.668 (0.641,0.694)
Fold 4	0.500	0.465 (0.440,0.493)	0.464	0.463 (0.438,0.489)	**0.750**	0.698 (0.673,0.724)	0.464	0.466 (0.439,0.492)
Fold 5	0.643	0.626 (0.596,0.657)	**0.786**	0.675 (0.640,0.713)	0.690	0.543 (0.503,0.583)	0.595	0.513 (0.482,0.543)
Fold 6	**0.967**	0.907 (0.889,0.923)	0.933	0.822 (0.793,0.848)	0.900	0.742 (0.705,0.777)	0.800	0.729 (0.705,0.754)
AIP	PAN-3		PAN-6		PAN-9		ResNet-18	
	Volume	Plane	Volume	Plane	Volume	Plane	Volume	Plane
Fold 1	0.738	0.643 (0.610,0.671)	**0.857**	0.780 (0.755,0.807)	0.786	0.753 (0.727,0.779)	0.786	0.705 (0.677, 0.735)
Fold 2	0.727	0.684 (0.658,0.708)	**0.818**	0.774 (0.752,0.796)	0.667	0.728 (0.704,0.750)	0.818	0.749 (0.725,0.774)
Fold 3	0.667	0.614 (0.584,0.642)	0.633	0.648 (0.623,0.671)	**0.683**	0.707 (0.682,0.729)	0.617	0.609 (0.580,0.636)
Fold 4	**0.982**	0.863 (0.848,0.879)	0.891	0.798 (0.773,0.820)	0.836	0.829 (0.810,0.848)	0.855	0.789 (0.767,0.810)
Fold 5	0.864	0.768 (0.738,0.794)	0.864	0.803 (0.777,0.828)	**0.909**	0.836 (0.815,0.857)	0.833	0.808 (0.782,0.831)
Fold 6	**0.900**	0.828 (0.810,0.846)	0.850	0.787 (0.766,0.806)	0.867	0.768 (0.746,0.788)	0.767	0.735 (0.711,0.758)

Table 8.6: AUC of class *previa* achieved by PAN-3, PAN-6 and PAN-9.

AUC	PAN-3		PAN-6		PAN-9	
	Volume	Plane	Volume	Plane	Volume	Plane
Fold 1	0.806	0.697 (0.673,0.722)	0.861	0.727 (0.703,0.750)	0.833	0.725 (0.702,0.749)
Fold 2	0.861	0.844 (0.827,0.862)	0.833	0.817 (0.798,0.836)	0.917	0.921 (0.909,0.933)
Fold 3	0.848	0.821 (0.800,0.841)	0.879	0.860 (0.842,0.876)	0.909	0.905 (0.891,0.919)
Fold 4	0.833	0.712 (0.690,0.736)	0.625	0.582 (0.556,0.606)	0.729	0.665 (0.642,0.689)
Fold 5	0.864	0.806 (0.785,0.827)	0.909	0.882 (0.867,0.897)	0.879	0.799 (0.779,0.818)
Fold 6	0.871	0.806 (0.784,0.827)	0.886	0.791 (0.769,0.813)	0.686	0.637 (0.612,0.663)

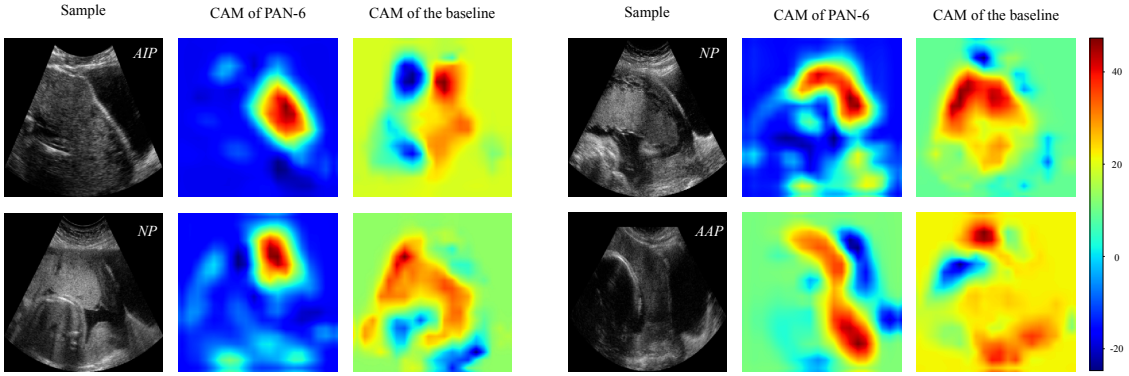


Figure 8.11: CAM samples with PASD screen labels (white), generated from PAN-6 and baseline respectively.

activation mapping (CAM), a technique that visualizes discriminative regions of a given class in images. Figure 8.11 illustrates CAM samples generated by PAN-6 and the baseline model respectively. It is clear that the proposed collaborative mechanism in PAN helps it focus on the clinically relevant regions while the baseline model has to learn such behaviour out of raw data in a less efficient way. This may lead to the sub-optimal performance of the baseline model compared to the proposed pipeline. For the auxiliary task, placenta previa assessment, Table 8.6 displays volume-wise AUCs achieved by PAN-3, PAN-6 and PAN-9 for class *previa*, which is of most clinical interest.

8.3 Comparison to Human Experts

It is intuitive to compare the proposed image-based prenatal assessment pipeline against human experts. To this end, a panel of thirteen fetal medicine experts were invited to participate in an online test. Due to the fact that there is no recognised training programme for PASD diagnosis, it is difficult to officially guarantee expertise.

However, the panel members are all experienced in PASD[4] and regularly perform prenatal ultrasound scans to diagnose PASD. Particularly, they were invited to independently classify all the placental ultrasound volumes in the test splits by visual inspection alone. Due to time constraints, each expert only assessed one test split except for one who examined all six. Each test split was assessed by three experts. For each test split, classification results from the three experts were recorded, along with their consensus[5]. Agreement between the PASD screen labels (i.e. gold standard from histopathology) and assessment results made by each expert as well as their consensus were measured using Cohen's Kappa coefficient. The fold-wise results are displayed in Table 8.7, which shows that PASD assessment from ultrasound images alone is essentially a very challenging task, causing strong inter-rater disagreements and low Cohen's Kappa coefficients between expert assessments and gold standards.

Plane-wise ROC curves of examined models (PAN-3, PAN-6, PAN-9, the baseline), along with expert performance, are shown in Figure 8.12, Figure 8.13 and Figure 8.14 for each fold and each PASD class respectively. Following conventions from previous work [214, 216], this thesis reports an expert's performance in the form of a sensitivity-specificity point. A machine-learning model achieves superior performance to an expert if the sensitivity-specificity point of the expert lies below the model's ROC curve. These results lead to the following class-wise observations and arguments:

- **Class NP**. Three PAN models have similar performances across six folds. Each achieves best volume-wise AUC scores in two folds. This indicates that the extrapolation distance D_{upi} is not a discriminative feature for NP classification. This result is in line with clinical guidelines: NP cases generally have visually distinct boundaries between the placenta and myometrium, which is a strong feature itself. The model thus cannot further benefit from a larger D_{upi}. In Fold 1, PAN-6 and PAN-9 outperforms all experts according

[4]All of the experts are members of the Internation Society for Abnormally Invasive Placenta (IS-AIP).

[5]In circumstances where three experts voted for three different classes, preference went to the expert who examined all six test splits.

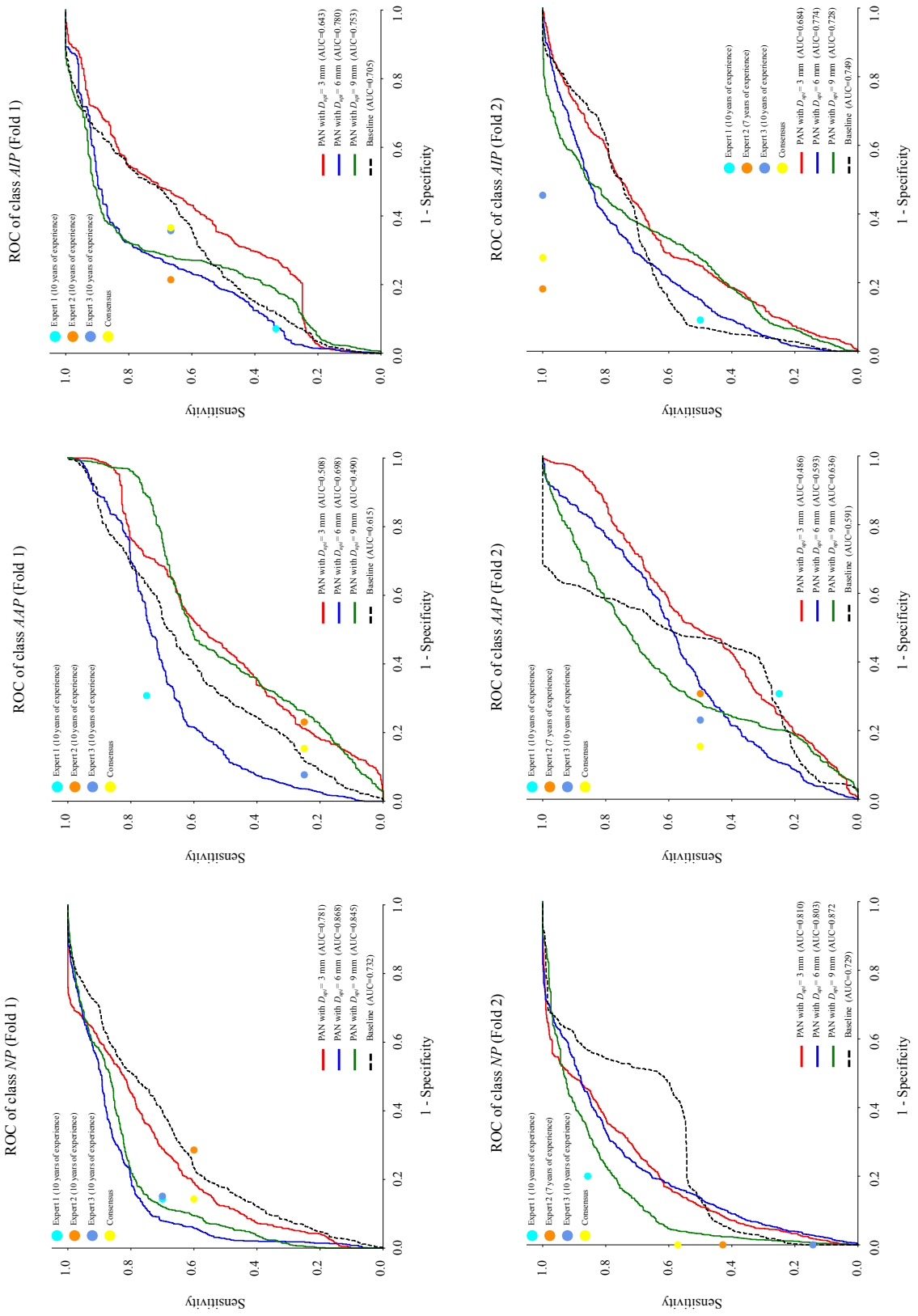


Figure 8.12: Plane-wise ROC curves for Fold 1 and Fold 2.

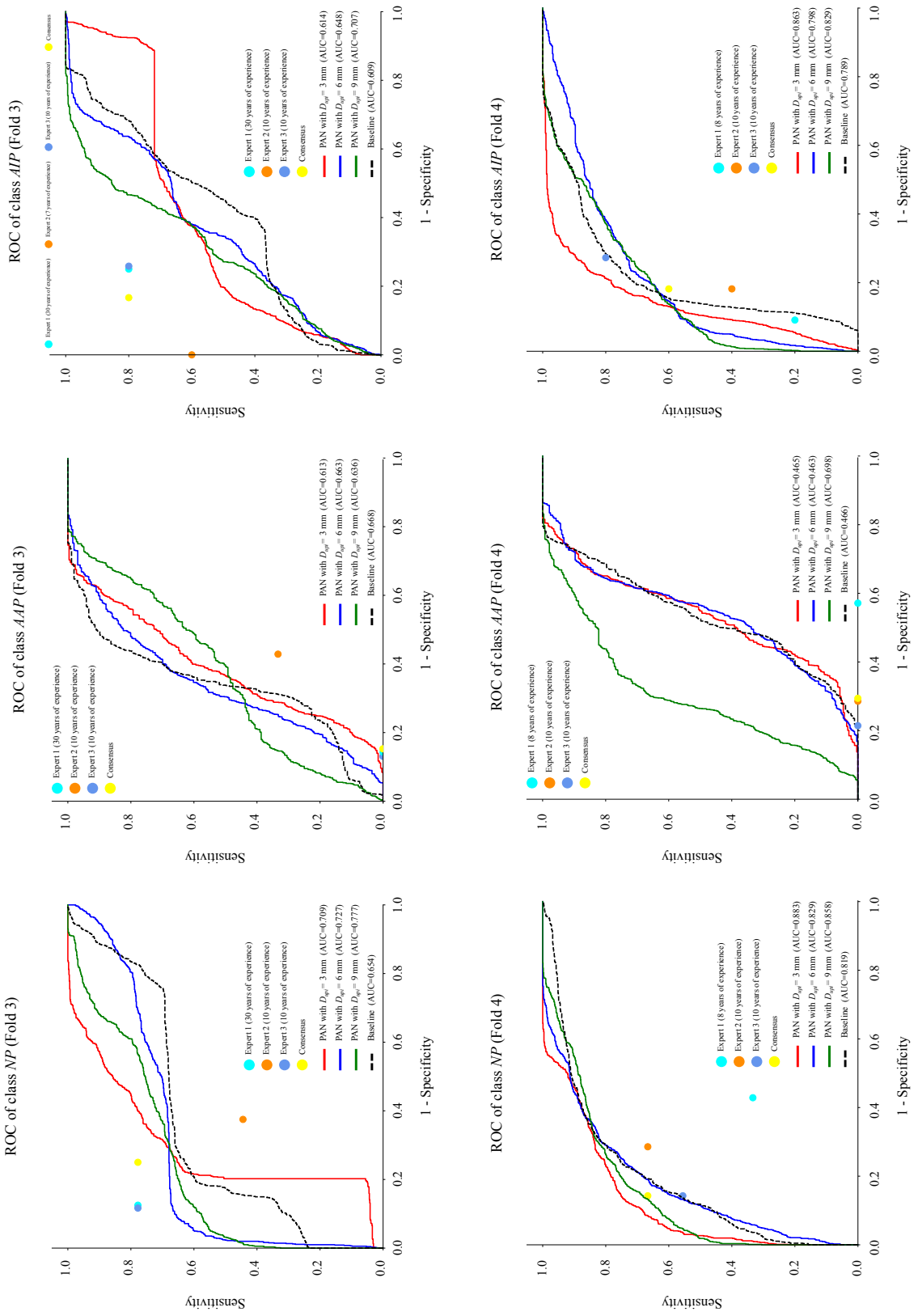


Figure 8.13: Plane-wise ROC curves for Fold 3 and Fold 4.

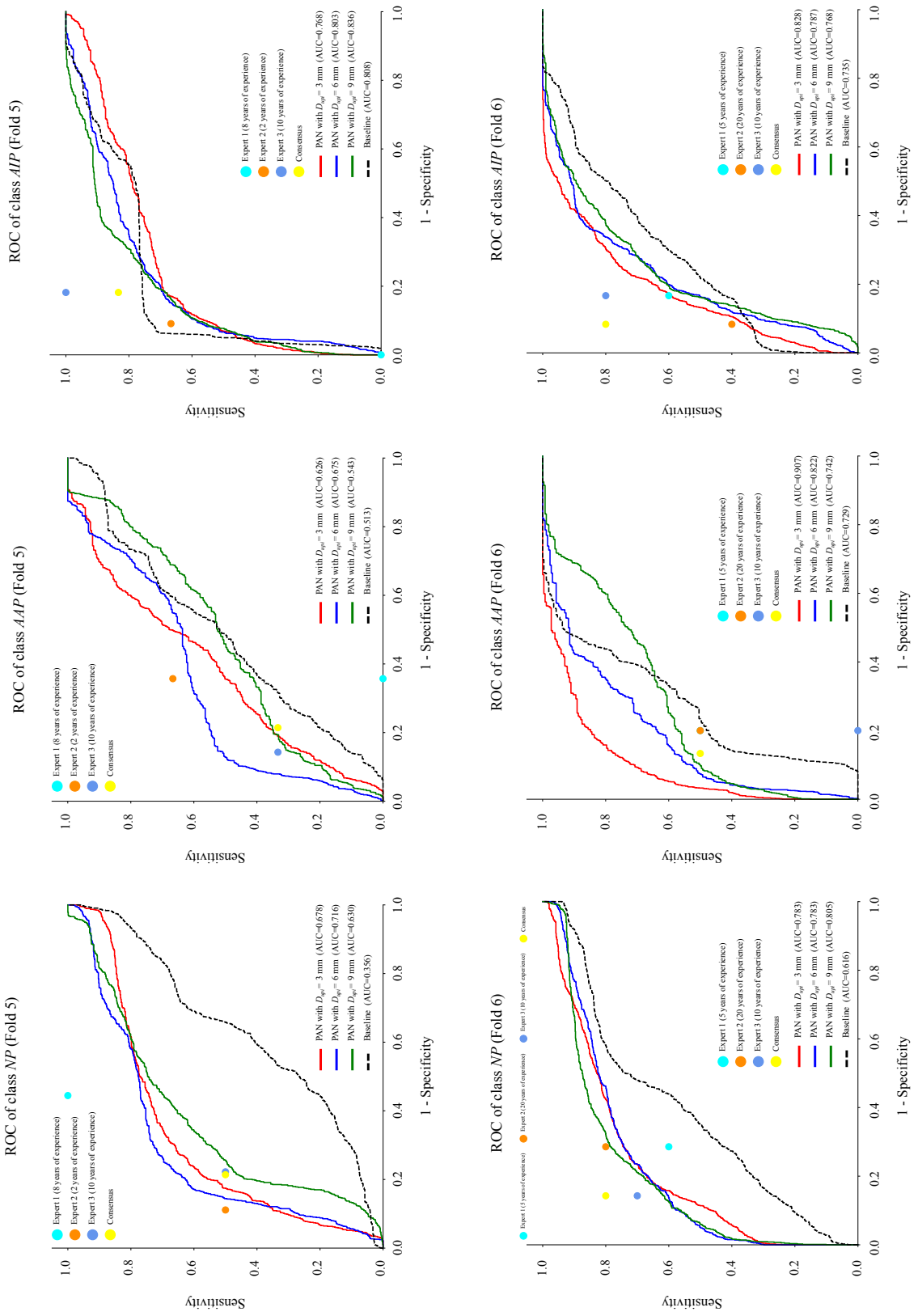


Figure 8.14: Plane-wise ROC curves for Fold 5 and Fold 6.

to their sensitivity-specificity points. Similarly in Fold 4, PAN-3 and PAN-9 outperforms all experts. In Fold 2, experts are better than PAN models. In general, PAN models perform on par with experts in classifying NP cases. From experts' perspective, these results may be attributed to experts' inherent bias towards abnormal cases, which can sometimes lead to over-diagnosis since the desire for high sensitivity (i.e. the ability to correctly detect all adverse cases) is usually regarded as more important than the specificity (i.e. the ability to correctly reject healthy cases) in the clinical diagnosis of this potentially life-threatening condition.

- **Class AAP**. PAN-6 achieves best performance in four out of six folds, which have two implications. First, setting the extrapolation distance to 6 mm is well-suited for AAP classification as sufficient PASD-related features in this neighbourhood are incorporated in a model's decision-making process. Second, AUC scores and experts' sensitivity / specificity scores are the lowest in AAP compared to scores in other two classes. In Fold 3 and Fold 4, several experts even had zero sensitivity. Anecdotally, experts find it most difficult to diagnose AAP, mainly due to the subtlety of AAP's sonographic signs. The PAN models show promising results by outperforming almost all experts in terms of classifying these AAP cases.

- **Class AIP**. Similar to NP, classifying AIP is relatively easier for experts and PAN models in terms of evaluation scores. In particular, each PAN model achieves best volume-wise in two folds, which indicates that D_{upi} may not be a discriminative feature for AIP classification. PAN models achieve similar performance to experts in classifying AIP cases. Strong sonographic signs such as numerous irregular PLs or UPI bulge are usually indicative of AIP, which suggests that the successful classification of AIP may benefit from the proposed collaborative mechanism of PAN.

Table 8.7: Cohen's Kappa statistics. E1, E2 and E3 denote three experts who participated in certain folds. Note that E1 and E2 refer to different experts across folds while E3 refers to the expert who examined all six test splits. GS denotes the histopathological gold standard.

Fold 1	E1	E2	E3	Consensus
GS	0.417	0.236	0.339	0.269
E1		0.545	0.482	0.571
E2			0.720	0.817
E3				0.904

Fold 2	E1	E2	E3	Consensus
GS	0.370	0.480	0.306	0.555
E1		0.246	0.038	0.323
E2			0.393	0.812
E3				0.590

Fold 3	E1	E2	E3	Consensus
GS	0.420	0.195	0.420	0.407
E1		0.313	0.605	0.702
E2			0.399	0.474
E3				0.901

Fold 4	E1	E2	E3	Consensus
GS	-0.091	0.179	0.304	0.300
E1		0.143	0.209	0.349
E2			0.524	0.707
E3				0.808

Fold 5	E1	E2	E3	Consensus
GS	0.140	0.399	0.446	0.364
E1		0.036	0.074	0.137
E2			0.485	0.566
E3				0.908

Fold 6	E1	E2	E3	Consensus
GS	0.324	0.386	0.410	0.595
E1		0.236	0.536	0.717
E2			0.240	0.407
E3				0.809

8.4 Conclusions

This chapter has presented a novel image-based prenatal assessment pipeline for PASD using a relatively small placental ultrasound dataset. In the first stage of the pipeline, an informative plane screening network was trained to filter out non-informative sagittal planes, which achieved high AUCs across text splits of a 6-fold nested cross-validation training / testing protocol. The second stage applied PL localization and UPI detection models from previous chapters to perform plane-wise pattern recognition with the purpose to extract PASD-related ultrasound signs. The final stage conducted collaborative PASD assessment, which integrated results from previous stages using the proposed PAN model. Furthermore, a panel of experts specialized in PASD were invited for independent assessment of samples from test splits. Experimental results showed that the proposed assessment pipeline matches the performance of experts at diagnosing *NP* and *AIP*, while outperformed experts at diagnosing *AAP*. These promising results suggested that the proposed screening pipeline has the potential to assist clinicians in the prenatal diagnosis of PASD by providing reliable second opinions as well as intermediate results of PASD-related sonographic pattern recognition.

Results from this chapter also imply that designing non-end-to-end application-specific deep-learning models can be an effective approach to challenges such as data shortage and lack of model explainability, which hinder the wide adoption of deep-learning algorithms in medical practice. It is expected that this would provide some insight for future work that may confront similar challenges in other clinical scenarios.

9 Conclusions and Future Work

Contents

9.1 Summary of Contributions 169
9.2 Directions for Future Work 173
9.2.1 Three-dimensional Sonographic Pattern Recognition . . 173
9.2.2 Joint Analysis of B-mode and Doppler Ultrasound . . . 173

9.1 Summary of Contributions

This thesis has developed and evaluated an machine-learning-based algorithmic pipeline for prenatal assessment of placenta accreta spectrum disorders (PASD) from three-dimensional placental ultrasound volumes. Although exceptionally rare, PASD drastically affect maternal health. A cross-institutional placental ultrasound image analysis database, namely the Oxford-Hackensack database of PASD, has been collected and annotated to validate the algorithms in this work. The proposed pipeline adopts a three-stage workflow to enhance model explainability. Overall, it consists of customized computational modules that carry out effective data pruning, extract PASD-related sonographic patterns from placental ultrasound images and make probabilistic estimates of PASD screen labels, which include the *normal placenta* (NP), *abnormally adherent placenta* (AAP) and *abnormally*

invasive placenta (AIP), ranked according to an increasing degree of placental invasion into the uterine wall. For evaluation purpose only, a panel of experts who specialize in PASD diagnosis and actively perform prenatal ultrasound scans were invited to independently assess test data. Specifically, chapter-wise contributions are summarized as follows:

- Chapter 3 presented the Oxford-Hackensack database of PASD. It started by introducing the ultrasound scanning protocol, based on which these 101 3D B-mode ultrasound samples were collected from John Radcliffe Hospital (Oxford, United Kingdom) and Hackensack University Medical Center (New Jersey, United States). The major contribution made by the author of this thesis with respect to the database was to expand it by adding a variety of PASD-related annotations to complement the original PASD screen label from histopathological examinations. First, all volumes were sliced along the sagittal planes to obtain 58,202 image planes. A binary image information label was tagged to each plane to describe whether it clearly depicts a list of anatomical structures without obvious truncation, thus is informative enough for PASD assessment. Each informative plane was then labelled with placental lacunae (PL) dot annotations, which recorded spatial coordinates of the centroid of identified PLs. The utero-placental interface (UPI) of each informative plane was delineated as a curve of 1-pixel width and stored as a binary mask. Finally, a placental previa screen label was tagged to each informative plane to categorize it as *no sign of previa*, *low-lying placenta*, or *placenta previa*.

- Chapter 4 introduced a method for performing weakly supervised localization of anatomical structures in placental ultrasound images using only image-level labels. Several novel contributions were made to facilitate the development of this method. First, a patch-based multi-structure categorization was proposed that enabled a deep residual network to construct discriminative features for classification. Second, the use of class activation mapping (CAM) technique

can visualize class-wise discriminative regions thus achieve weakly supervised localization without using any pixel-level annotation.

- Chapter 5 introduced multi-scale aggregation methods for PL localization in placental ultrasound images, which is a key sonographic pattern recognition task related to PASD assessment. Several technical contributions were made in this chapter to develop a principled approach to PL localization and error analysis. First, a method was presented to transform dot annotations to local confidence maps based on low-level over-segmentation technique. Two different multi-scale feature aggregation methods were evaluated: one with a feature pyramid structure that aggregated from the deepest stage to the shallowest stage, and a second with an iterative deep aggregation structure that aggregated from the shallowest stage to the deepest stage. Experimental validation suggested that the second method outperformed the first one since its preference towards shallower stages facilitated PL representation learning. Finally, a detailed error analysis framework was presented to investigate the cause of PL localization errors.

- Chapter 6 introduced a method for detecting UPI in placental ultrasound images, which is another key sonographic pattern recognition task related to PASD assessment since placental adherence or invasion occurs along UPI. Accurate detection of UPI would clearly provide strong spatial cues to guide PASD-related representation learning. The most notable contribution made in this chapter was to formulate UPI detection as a semantic edge detection problem and address it by exploiting global context modelling to filter out unwanted false postive predictions associated to UPI. The resulting *UPI-Net* model was shown to give better performance for the challenging task of UPI detection compared to several competitive benchmark models.

- Chapter 7 introduced a method to extend the proposed UPI-Net in Chapter 6 for the task of UPI detection in placental ultrasound images. It was based on the observation that failure cases made by UPI-Net generally lacked spatial

continuity and crispness. The major contribution made in this chapter was to introduce two auxiliary tasks to regularize the network's behaviours without using any extra manual annotation, i.e. the supervision signals used in the auxiliary tasks were directly derived from the original binary UPI reference masks. The first auxiliary task was to enforce spatial continuity for UPI predictions using a conditional generative adversarial learning framework. The corresponding supervision signals were Gaussian density maps derived from UPI binary masks via distance transform. The second auxiliary task was to segment gradient-sensitive masks, which were also derived from UPI binary masks and local image derivatives. Experimental results suggested that the proposed *UPI-Net++* model benefited from the auxiliary tasks in terms of evaluation metrics and qualitative examinations.

- Chapter 8 presented the image-based prenatal assessment pipeline for PASD, which adopted a three-stage workflow to integrate computational modules in a novel way. The first stage conducted informative plane screening using a deep residual network. The resulting model achieved more than 0.9 AUCs across six non-overlapping test splits of the database. The second stage performed PL localization and UPI detection to obtain these key sonographic patterns related to PASD. The third stage presented a PASD assessment network that integrated image planes with their estimated PL heatmaps and UPI masks in a collaborative manner to facilitate PASD-related representation learning. Experimental results revealed that the proposed PASD assessment pipeline matches the performance of experts at classifying NP and AIP cases while outperformed experts in classifying AAP cases, which showed that the pipeline has the potential to assist clinicians in prenatal assessment for PASD by providing reliable second opinions and visualizing ultrasound signs related to the condition.

9.2 Directions for Future Work

This section broadly discusses two directions for future work enabled by this thesis: three-dimensional sonographic pattern recognition and joint analysis of B-mode and Doppler ultrasound.

9.2.1 Three-dimensional Sonographic Pattern Recognition

Chapter 5 presents a 2D approach for PL localization, which treats each image plane independently during training and testing. In practice, this leads to reasonably good localization performance in terms of average precision metrics by matching predicted PL instances with reference instances (see § 5.4). To some extent, 2D approaches compensate for the lack of training data by cutting one dimension to alleviate the well-known curse of dimensionality [227]. However, since PLs are mostly caused by hyper-vascular activities inside placenta, they have 3D characteristics by nature. Instead of treating PLs that are spatially close to each other along sagittal planes as independent instances, the possibility that they belong to a common (3D) PL should be considered and checked either by algorithmic approaches such as 3D over-segmentation techniques [138] or by vasculature examinations using Doppler ultrasound. Similar to § 5.2, a 3D Gaussian density map can be fitted to identified 3D PL for subsequent analysis using machine learning algorithms such as 3D fully convolutional neural network. Based on this thesis, a series of 3D sonographic pattern recognition methods can be developed for UPI detection, placental previa classification and 3D PASD assessment.

9.2.2 Joint Analysis of B-mode and Doppler Ultrasound

First, it is worth noting that all experiments and analysis are based on a relatively small dataset of an exceptionally rare pregnancy complication. Although promising results have been achieved based on a carefully designed pipeline, it is without doubt that increasing the amount of training data should be the easiest way to improve PASD assessment performance, while increasing the amount of test data may lead to conclusions that are more statistically sound. Second, this thesis focuses on B-mode

placental ultrasound scans alone for prenatal assessment of PASD. However, experts tend to rely on not only B-mode scans but also placental vascularity (e.g. from Doppler ultrasound) and patients' medical history to make diagnostic decisions. This is likely to lead to higher sensitivity and specificity than those shown in § 8.3.

Intuitively, integrating these pieces of information into machine-learning models would be a natural extension of this thesis. Power Doppler ultrasound is often used to visualize blood perfusion by measuring the amplitude of scattering of ultrasound waves. However, this unstandardised measure is affected by a large number of factors including effects of attenuation (maternal habitus, fetal depth, intervening tissue type) and machine settings (aperture size, gain, scale, filter and persistence). Traditional ways to quantify power Doppler signals involve vascular indices, which has been controversial due to the standardization problem [228–230]. One promising effort to resolve this issue leads to a signal normalization method called fractional moving blood volume (FMBV) [229, 231]. It works by first choosing a maximum reference value and assigning 100% to vascular amplitude scatterings of this value. The rest regional measurements are rescaled as fractional perfusion of the reference point. Investigating placental vascularity associated with PASD may help identify structural and vascular biomarkers or sonographic patterns for accurate prenatal diagnosis. Therefore, an interesting future direction is to carry out joint analysis of B-mode ultrasound and FMBV for PASD assessment.

Appendices

A Appendix

Contents

A.1 Implementation of Algorithms 176

A.1 Implementation of Algorithms

In this thesis, all algorithms were implemented by the author in the Python programming language using the following open source libraries:

- PyTorch 1.2 (`https://pytorch.org/`): deep learning framework with CPU/GPU supports.
- imgaug 0.2.9 (`https://imgaug.readthedocs.io/en/latest/`): data augmentation for training deep-learning models.
- matplotlib 3.1.0 (`https://matplotlib.org/`): basic plotting.
- numpy 1.16.2 (`https://numpy.org/`): basic linear algebra operations.
- scikit-image 0.15.0 (`https://scikit-image.org/`): basic image processing.

- scikit-learn 0.21.2 (`https://scikit-learn.org/stable/`): machine learning framework.
- SimpleITK 1.2.0 (`http://www.simpleitk.org/`): reading 3D ultrasound volumes.
- torchvision 0.3.0 (`https://pytorch.org/docs/stable/torchvision/index.html`): a deep learning model zoo with ImageNet-pretrained parameters available.
- tensorboardX 1.8 (`https://tensorboardx.readthedocs.io/en/latest/index.html`): logging tools for running deep-learning models.

References

[1] Mercy PrabhuDas et al. “Immune mechanisms at the maternal-fetal interface: perspectives and challenges”. In: *Nature immunology* 16.4 (2015), pp. 328–334.

[2] Yuping Wang. “Vascular biology of the placenta”. In: *Colloquium Series on Integrated Systems Physiology: From Molecule to Function*. Vol. 2. 1. Morgan & Claypool Life Sciences. 2010, pp. 1–98.

[3] JM Roberts and Desmond W Cooper. “Pathogenesis and genetics of pre-eclampsia”. In: *The Lancet* 357.9249 (2001), pp. 53–56.

[4] AA Baschat. “Neurodevelopment following fetal growth restriction and its relationship with antepartum parameters of placental dysfunction”. In: *Ultrasound in obstetrics & gynecology* 37.5 (2011), pp. 501–514.

[5] Christiane Krebs et al. “Intrauterine growth restriction with absent end-diastolic flow velocity in the umbilical artery is associated with maldevelopment of the placental terminal villous tree”. In: *American journal of obstetrics and gynecology* 175.6 (1996), pp. 1534–1542.

[6] RM Lewis, JK Cleal, and Mark A Hanson. “Placenta, evolution and lifelong health”. In: *Placenta* 33 (2012), S28–S32.

[7] David JP Barker et al. “The placental origins of sudden cardiac death”. In: *International journal of epidemiology* 41.5 (2012), pp. 1394–1399.

[8] David JP Barker and Kent L Thornburg. “Placental programming of chronic diseases, cancer and lifespan: a review”. In: *Placenta* 34.10 (2013), pp. 841–845.

[9] Alan E Guttmacher, Yvonne T Maddox, and Catherine Y Spong. “The Human Placenta Project: placental structure, development, and function in real time”. In: *Placenta* 35.5 (2014), pp. 303–304.

[10] Helen Kay, MD D Michael Nelson, and Yuping Wang. *The placenta: from development to disease*. John Wiley & Sons, 2011.

[11] Lawrence W Oppenheimer et al. “What is a low-lying placenta?” In: *American journal of obstetrics and gynecology* 165.4 (1991), pp. 1036–1038.

[12] Amarnath Bhide et al. “Placental edge to internal os distance in the late third trimester and mode of delivery in placenta praevia”. In: *BJOG: An International Journal of Obstetrics & Gynaecology* 110.9 (2003), pp. 860–864.

[13] William B Dawson et al. “Translabial ultrasonography and placenta previa: does measurement of the os-placenta distance predict outcome?” In: *Journal of ultrasound in medicine* 15.6 (1996), pp. 441–446.

[14] Eric Jauniaux, Sally Collins, and Graham J Burton. “Placenta accreta spectrum: pathophysiology and evidence-based anatomy for prenatal ultrasound imaging”. In: *American journal of obstetrics and gynecology* 218.1 (2018), pp. 75–87.

[15] Yinka Oyelese and John C Smulian. "Placenta previa, placenta accreta, and vasa previa". In: *Obstetrics & Gynecology* 107.4 (2006), pp. 927–941.

[16] Solomon Iyasu et al. "The epidemiology of placenta previa in the United States, 1979 through 1987". In: *American journal of obstetrics and gynecology* 168.5 (1993), pp. 1424–1429.

[17] Kurt Benirschke, Graham J Burton, and Rebecca N Baergen. "Early development of the human placenta". In: *Pathology of the human placenta*. Springer, 2012, pp. 41–53.

[18] RH Becker et al. "The relevance of placental location at 20–23 gestational weeks for prediction of placenta previa at delivery: evaluation of 8650 cases". In: *Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology* 17.6 (2001), pp. 496–501.

[19] L Thurn et al. "Abnormally invasive placenta - prevalence, risk factors and antenatal suspicion: results from a large population-based pregnancy cohort study in the Nordic countries". In: *BJOG* 123.8 (2016), pp. 1348–1355.

[20] Michal Kovo, Letizia Schreiber, and Jacob Bar. "Placental vascular pathology as a mechanism of disease in pregnancy complications". In: *Thrombosis research* 131 (2013), S18–S21.

[21] R Kelly et al. "Placental vascular pathology findings and pathways to preterm delivery". In: *American journal of epidemiology* 170.2 (2009), pp. 148–158.

[22] Eric Jauniaux et al. "FIGO consensus guidelines on placenta accreta spectrum disorders: Prenatal diagnosis and screening". In: *International Journal of Gynecology & Obstetrics* 140.3 (2018), pp. 274–280.

[23] Cher Heng Tan et al. "Perioperative endovascular internal iliac artery occlusion balloon placement in management of placenta accreta". In: *American Journal of Roentgenology* 189.5 (2007), pp. 1158–1163.

[24] Tamar Eshkoli et al. "Placenta accreta: risk factors, perinatal outcomes, and consequences for subsequent births". In: *American journal of obstetrics and gynecology* 208.3 (2013), 219–e1.

[25] Gali Garmi and Raed Salim. "Epidemiology, etiology, diagnosis, and management of placenta accreta". In: *Obstetrics and gynecology international* 2012 (2012).

[26] George Daskalakis et al. "Emergency obstetric hysterectomy". In: *Acta obstetricia et gynecologica Scandinavica* 86.2 (2007), pp. 223–227.

[27] Adi Orbach et al. "Peripartum cesarean hysterectomy: critical analysis of risk factors and trends over the years". In: *The Journal of Maternal-Fetal & Neonatal Medicine* 24.3 (2011), pp. 480–484.

[28] John M O'Brien, John R Barton, and Elvis S Donaldson. "The management of placenta percreta: conservative and operative strategies". In: *American journal of obstetrics and gynecology* 175.6 (1996), pp. 1632–1638.

[29] Tom Rosenberg et al. "Critical analysis of risk factors and outcome of placenta previa". In: *Archives of gynecology and obstetrics* 284.1 (2011), pp. 47–51.

[30] Serena Wu, Masha Kocherginsky, and Judith U Hibbard. "Abnormal placentation: twenty-year analysis". In: *American journal of obstetrics and gynecology* 192.5 (2005), pp. 1458–1461.

[31] David A Miller, Janet A Chollet, and T Murphy Goodwin. "Clinical risk factors for placenta previa–placenta accreta". In: *American journal of obstetrics and gynecology* 177.1 (1997), pp. 210–214.

[32] Ilan E Timor-Tritsch and Ana Monteagudo. "Unforeseen consequences of the increasing rate of cesarean deliveries: early placenta accreta and cesarean scar pregnancy. A review". In: *American journal of obstetrics and gynecology* 207.1 (2012), pp. 14–29.

[33] Donal J Brennan et al. "Comparative analysis of international cesarean delivery rates using 10-group classification identifies significant variation in spontaneous labor". In: *American journal of obstetrics and gynecology* 201.3 (2009), 308–e1.

[34] Francis C Notzon. "International differences in the use of obstetric interventions". In: *Jama* 263.24 (1990), pp. 3286–3291.

[35] Jacques Balayla and Helen Davis Bondarenko. "Placenta accreta and the risk of adverse maternal and neonatal outcomes". In: *Journal of perinatal medicine* 41.2 (2013), pp. 141–149.

[36] Karla N Solheim et al. "The effect of cesarean delivery rates on the future incidence of placenta previa, placenta accreta, and maternal mortality". In: *The Journal of Maternal-Fetal & Neonatal Medicine* 24.11 (2011), pp. 1341–1346.

[37] Frédéric Chantraine et al. "Prenatal diagnosis of abnormally invasive placenta reduces maternal peripartum hemorrhage and morbidity". In: *Acta obstetricia et gynecologica Scandinavica* 92.4 (2013), pp. 439–444.

[38] Hong Soo Wong et al. "Evaluation of sonographic diagnostic criteria for placenta accreta". In: *Journal of Clinical Ultrasound* 36.9 (2008), pp. 551–559.

[39] JC Shih et al. "Role of three-dimensional power Doppler in the antenatal diagnosis of placenta accreta: comparison with gray-scale and color Doppler techniques". In: *Ultrasound in obstetrics and gynecology* 33.2 (2009), pp. 193–203.

[40] Sally L Collins et al. "Three-dimensional power Doppler ultrasonography for diagnosing abnormally invasive placenta and quantifying the risk". In: *Obstetrics and gynecology* 126.3 (2015), p. 645.

[41] Christine H Comstock et al. "Sonographic detection of placenta accreta in the second and third trimesters of pregnancy". In: *American journal of obstetrics and gynecology* 190.4 (2004), pp. 1135–1140.

[42] D Levine et al. "Placenta accreta: evaluation with color Doppler US, power Doppler US, and MR imaging." In: *Radiology* 205.3 (1997), pp. 773–776.

[43] Sally Collins et al. "Proposal for standardized ultrasound descriptors of abnormally invasive placenta (AIP)". In: *Ultrasound in Obstetrics & Gynecology* 47.3 (2016), pp. 271–275.

[44] Frederic Chantraine and Sally L Collins. "Prenatal Ultrasound Imaging for Placenta Accreta Spectrum (PAS): a Practical Guide". In: *Current Obstetrics and Gynecology Reports* 8.3 (2019), pp. 86–93.

[45] Stewart F Cramer et al. "Placenta Accreta and Placenta Increta: An Approach to Pathogenesis Based on the Trophoblastic Differentiation Pathway". In: *Pediatric and Developmental Pathology* 19.4 (2016), pp. 320–333.

[46] Dorota A Doherty et al. “Pre-pregnancy body mass index and pregnancy outcomes”. In: *International Journal of Gynecology & Obstetrics* 95.3 (2006), pp. 242–247.

[47] HJ Finberg et al. “Placenta accreta: prospective sonographic diagnosis in patients with placenta previa and prior cesarean section”. In: *Journal of ultrasound in medicine* 11.7 (1992), pp. 333–343.

[48] Myron A Pozniak, James A Zagzebski, and Kathleen A Scanlan. “Spectral and color Doppler artifacts.” In: *Radiographics* 12.1 (1992), pp. 35–44.

[49] Eli Gibson et al. “Inter-site Variability in Prostate Segmentation Accuracy Using Deep Learning”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 506–514.

[50] SL Collins et al. “Influence of power Doppler gain setting on Virtual Organ Computer-aided AnaLysis indices in vivo: can use of the individual sub-noise gain level optimize information?” In: *Ultrasound in Obstetrics & Gynecology* 40.1 (2012), pp. 75–80.

[51] SL Collins et al. “Developmental changes in spiral artery blood flow in the human placenta observed with colour Doppler ultrasonography”. In: *Placenta* 33.10 (2012), pp. 782–787.

[52] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444.

[53] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[54] Ronan Collobert et al. “Natural language processing (almost) from scratch”. In: *Journal of machine learning research* 12.Aug (2011), pp. 2493–2537.

[55] Geoffrey Hinton et al. “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups”. In: *IEEE Signal processing magazine* 29.6 (2012), pp. 82–97.

[56] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540 (2015), p. 529.

[57] David Silver et al. “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529.7587 (2016), p. 484.

[58] David Silver et al. “Mastering the game of go without human knowledge”. In: *Nature* 550.7676 (2017), p. 354.

[59] Pierre Baldi, Peter Sadowski, and Daniel Whiteson. “Searching for exotic particles in high-energy physics with deep learning”. In: *Nature communications* 5 (2014), p. 4308.

[60] Michael KK Leung et al. “Deep learning of the tissue-regulated splicing code”. In: *Bioinformatics* 30.12 (2014), pp. i121–i129.

[61] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[62] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[63] David H Hubel and Torsten N Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex". In: *The Journal of physiology* 160.1 (1962), pp. 106–154.

[64] Daniel J Felleman and DC Essen Van. "Distributed hierarchical processing in the primate cerebral cortex." In: *Cerebral cortex (New York, NY: 1991)* 1.1 (1991), pp. 1–47.

[65] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. "Rectifier nonlinearities improve neural network acoustic models". In: *Proceedings of the international conference on Machine learning*. Vol. 30. 1. 2013, p. 3.

[66] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *arXiv preprint arXiv:1502.03167* (2015).

[67] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. "Instance normalization: The missing ingredient for fast stylization". In: *arXiv preprint arXiv:1607.08022* (2016).

[68] Tim Salimans and Durk P Kingma. "Weight normalization: A simple reparameterization to accelerate training of deep neural networks". In: *Advances in Neural Information Processing Systems*. 2016, pp. 901–909.

[69] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. "Layer normalization". In: *arXiv preprint arXiv:1607.06450* (2016).

[70] Yuxin Wu and Kaiming He. "Group normalization". In: *Proceedings of the European Conference on Computer Vision*. 2018, pp. 3–19.

[71] Yann LeCun et al. "Handwritten digit recognition with a back-propagation network". In: *Advances in neural information processing systems*. 1990, pp. 396–404.

[72] Ilya Sutskever et al. "On the importance of initialization and momentum in deep learning". In: *International conference on machine learning*. 2013, pp. 1139–1147.

[73] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization". In: *Journal of machine learning research* 12.Jul (2011), pp. 2121–2159.

[74] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[75] Ilya Loshchilov and Frank Hutter. "Decoupled weight decay regularization". In: *arXiv preprint arXiv:1711.05101* (2017).

[76] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[77] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[78] Antreas Antoniou, Amos Storkey, and Harrison Edwards. "Data augmentation generative adversarial networks". In: *arXiv preprint arXiv:1711.04340* (2017).

[79] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[80] Zhe Cao et al. "Realtime multi-person 2d pose estimation using part affinity fields". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 7291–7299.

[81] Zenglin Shi, Pascal Mettes, and Cees GM Snoek. "Counting with focus for free". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 4200–4209.

[82] Jifeng Dai et al. "R-fcn: Object detection via region-based fully convolutional networks". In: *Advances in neural information processing systems*. 2016, pp. 379–387.

[83] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[84] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[85] Christian Szegedy et al. "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[86] Zhen Yu et al. "A deep convolutional neural network-based framework for automatic fetal facial standard plane recognition". In: *IEEE journal of biomedical and health informatics* 22.3 (2017), pp. 874–885.

[87] Seokmin Han et al. "A deep learning framework for supporting the classification of breast lesions in ultrasound images". In: *Physics in Medicine & Biology* 62.19 (2017), p. 7714.

[88] Lingyun Wu et al. "FUIQA: Fetal ultrasound image quality assessment with deep convolutional networks". In: *IEEE transactions on cybernetics* 47.5 (2017), pp. 1336–1349.

[89] Moi Hoon Yap et al. "Automated breast ultrasound lesions detection using convolutional neural networks". In: *IEEE journal of biomedical and health informatics* 22.4 (2017), pp. 1218–1226.

[90] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[91] Christian F Baumgartner et al. "SonoNet: real-time detection and localisation of fetal standard scan planes in freehand ultrasound". In: *IEEE transactions on medical imaging* 36.11 (2017), pp. 2204–2215.

[92] Xin Yang et al. "Fine-grained recurrent neural networks for automatic prostate segmentation in ultrasound images". In: *Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

[93] Emran Mohammad Abu Anas et al. "Clinical target-volume delineation in prostate brachytherapy using residual neural networks". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2017, pp. 365–373.

[94] Yi Wang et al. "Deep attentive features for prostate segmentation in 3d transrectal ultrasound". In: *IEEE transactions on medical imaging* 38.12 (2019), pp. 2768–2778.

[95] Pádraig Looney et al. "Automatic 3D ultrasound segmentation of the first trimester placenta using deep learning". In: *IEEE International Symposium on Biomedical Imaging*. IEEE. 2017, pp. 279–282.

[96] Konstantinos Kamnitsas et al. "DeepMedic for brain tumor segmentation". In: *International workshop on Brainlesion: Glioma, multiple sclerosis, stroke and traumatic brain injuries*. Springer. 2016, pp. 138–149.

[97] Konstantinos Kamnitsas et al. "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation". In: *Medical Image Analysis* 36 (2017), pp. 61–78.

[98] Hans J. Johnson et al. "The ITK Software Guide". In: (2013). *In press*.

[99] Will Schroeder and Ken Martin. "The Visualization Toolkit". In: *Prenatal diagnosis* (2006).

[100] Eric Jauniaux et al. "FIGO classification for the clinical diagnosis of placenta accreta spectrum disorders". In: *International Journal of Gynecology & Obstetrics* 146.1 (2019), pp. 20–24.

[101] Piotr Dollár and C Lawrence Zitnick. "Structured forests for fast edge detection". In: *Proceedings of the IEEE international conference on computer vision*. 2013, pp. 1841–1848.

[102] Huan Qi, Sally Collins, and Alison Noble. "Weakly supervised learning of placental ultrasound images with residual networks". In: *Annual Conference on Medical Image Understanding and Analysis*. Springer. 2017, pp. 98–108.

[103] Amber L Simpson et al. "A large annotated medical image dataset for the development and evaluation of segmentation algorithms". In: *arXiv preprint arXiv:1902.09063* (2019).

[104] Loris Bazzani et al. "Self-taught object localization with deep networks". In: *2016 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2016, pp. 1–9.

[105] Maxime Oquab et al. "Is object localization for free?-weakly-supervised learning with convolutional neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 685–694.

[106] Bolei Zhou et al. "Learning deep features for discriminative localization". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2921–2929.

[107] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps". In: *arXiv preprint arXiv:1312.6034* (2013).

[108] Kaiming He et al. “Identity mappings in deep residual networks”. In: *European conference on computer vision*. Springer. 2016, pp. 630–645.

[109] Olga Russakovsky et al. “Imagenet large scale visual recognition challenge”. In: *International journal of computer vision* 115.3 (2015), pp. 211–252.

[110] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4 (1989), pp. 541–551.

[111] Hao Chen et al. “VoxResNet: Deep Voxelwise Residual Networks for Volumetric Brain Segmentation”. In: *arXiv preprint arXiv:1608.05895* (2016).

[112] Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. “Torch7: A matlab-like environment for machine learning”. In: *BigLearn, Advances in Neural Information Processing Systems workshop*. CONF. 2011.

[113] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010, pp. 249–256.

[114] Yoshua Bengio, Patrice Simard, Paolo Frasconi, et al. “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE transactions on neural networks* 5.2 (1994), pp. 157–166.

[115] Ramprasaath R Selvaraju et al. “Grad-cam: Visual explanations from deep networks via gradient-based localization”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 618–626.

[116] Huan Qi, Sally Collins, and J Alison Noble. “Automatic lacunae localization in placental ultrasound images via layer aggregation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 921–929.

[117] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. “Posefix: Model-agnostic general human pose refinement network”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7773–7781.

[118] Leonid Pishchulin et al. “Deepcut: Joint subset partition and labeling for multi person pose estimation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4929–4937.

[119] Alejandro Newell, Zhiao Huang, and Jia Deng. “Associative embedding: End-to-end learning for joint detection and grouping”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 2277–2287.

[120] Yilun Chen et al. “Cascaded pyramid network for multi-person pose estimation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7103–7112.

[121] Alejandro Newell, Kaiyu Yang, and Jia Deng. “Stacked hourglass networks for human pose estimation”. In: *European conference on computer vision*. Springer. 2016, pp. 483–499.

[122] Alexander Toshev and Christian Szegedy. “Deeppose: Human pose estimation via deep neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1653–1660.

[123] Matteo Ruggero Ronchi and Pietro Perona. "Benchmarking and error diagnosis in multi-instance pose estimation". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 369–378.

[124] Alison M Pouch et al. "Fully automatic segmentation of the mitral leaflets in 3D transesophageal echocardiographic images using multi-atlas joint label fusion and deformable medial modeling". In: *Medical image analysis* 18.1 (2014), pp. 118–129.

[125] Bulat Ibragimov et al. "Segmentation of pathological structures by landmark-assisted deformable models". In: *IEEE transactions on medical imaging* 36.7 (2017), pp. 1457–1469.

[126] Hans J Johnson and Gary E Christensen. "Consistent landmark and intensity-based image registration". In: *IEEE transactions on medical imaging* 21.5 (2002), pp. 450–461.

[127] Karl Rohr et al. "Landmark-based elastic registration using approximating thin-plate splines". In: *IEEE Transactions on medical imaging* 20.6 (2001), pp. 526–534.

[128] Lin Yang et al. "Prediction based collaborative trackers (PCT): A robust and accurate approach toward 3D medical object tracking". In: *IEEE transactions on medical imaging* 30.11 (2011), pp. 1921–1932.

[129] Yefeng Zheng et al. "3D deep learning for efficient and robust landmark detection in volumetric data". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2015, pp. 565–572.

[130] Christian Payer et al. "Regressing heatmaps for multiple landmark localization using CNNs". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016, pp. 230–238.

[131] Jun Zhang, Mingxia Liu, and Dinggang Shen. "Detecting anatomical landmarks from limited medical imaging data using two-stage task-oriented deep neural networks". In: *IEEE Transactions on Image Processing* 26.10 (2017), pp. 4753–4764.

[132] Florin-Cristian Ghesu et al. "Multi-scale deep reinforcement learning for real-time 3D-landmark detection in CT scans". In: *IEEE transactions on pattern analysis and machine intelligence* 41.1 (2017), pp. 176–189.

[133] Thorsten Falk et al. "U-Net: deep learning for cell counting, detection, and morphometry". In: *Nature methods* 16.1 (2019), p. 67.

[134] Christof Angermueller et al. "Deep learning for computational biology". In: *Molecular systems biology* 12.7 (2016).

[135] Daniele Ravi et al. "Deep learning for health informatics". In: *IEEE journal of biomedical and health informatics* 21.1 (2016), pp. 4–21.

[136] Weidi Xie, J Alison Noble, and Andrew Zisserman. "Microscopy cell counting and detection with fully convolutional regression networks". In: *Computer methods in biomechanics and biomedical engineering: Imaging & Visualization* 6.3 (2018), pp. 283–292.

[137] Pavle Prentasc and Sven Loncaric. "Detection of exudates in fundus photographs using deep neural networks and anatomical landmark detection fusion". In: *Computer methods and programs in biomedicine* 137 (2016), pp. 281–292.

[138] Radhakrishna Achanta et al. “SLIC superpixels compared to state-of-the-art superpixel methods”. In: *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), pp. 2274–2282.

[139] Yuyin Zhou et al. “Deep supervision for pancreatic cyst segmentation in abdominal CT scans”. In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2017, pp. 222–230.

[140] Saining Xie and Zhuowen Tu. “Holistically-nested edge detection”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1395–1403.

[141] Tsung-Yi Lin et al. “Feature pyramid networks for object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[142] Fisher Yu et al. “Deep layer aggregation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2403–2412.

[143] Jesse Davis and Mark Goadrich. “The relationship between Precision-Recall and ROC curves”. In: *Proceedings of the 23rd international conference on Machine learning*. ACM. 2006, pp. 233–240.

[144] Simon Kohl et al. “A probabilistic u-net for segmentation of ambiguous images”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 6965–6975.

[145] Huan Qi, Sally Collins, and J Alison Noble. “UPI-Net: Semantic Contour Detection in Placental Ultrasound”. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2019, pp. 0–0.

[146] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification and scene analysis*. Vol. 3. Wiley New York, 1973.

[147] Jerry R. Fram and Edward S. Deutsch. “On the quantitative evaluation of edge detection schemes and their comparison with human performance”. In: *IEEE Transactions on Computers* 100.6 (1975), pp. 616–628.

[148] Pablo Arbelaez et al. “Contour detection and hierarchical image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 33.5 (2010), pp. 898–916.

[149] Jitendra Malik et al. “Contour and texture analysis for image segmentation”. In: *International journal of computer vision* 43.1 (2001), pp. 7–27.

[150] Vittorio Ferrari, Tinne Tuytelaars, and Luc Van Gool. “Object detection by contour segment networks”. In: *Proceedings of the European conference on computer vision*. Springer. 2006, pp. 14–28.

[151] Vittorio Ferrari et al. “Groups of adjacent contour segments for object detection”. In: *IEEE transactions on pattern analysis and machine intelligence* 30.1 (2007), pp. 36–51.

[152] John Canny. “A computational approach to edge detection”. In: *IEEE Transactions on pattern analysis and machine intelligence* 6 (1986), pp. 679–698.

[153] Djemel Ziou, Salvatore Tabbone, et al. “Edge detection techniques-an overview”. In: *Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii* 8 (1998), pp. 537–559.

[154] Josef Kittler. "On the accuracy of the Sobel edge detector". In: *Image and Vision Computing* 1.1 (1983), pp. 37–42.

[155] Ren Xiaofeng and Liefeng Bo. "Discriminatively trained sparse code gradients for contour detection". In: *Advances in neural information processing systems*. 2012, pp. 584–592.

[156] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. "Perceptual organization and recognition of indoor scenes from RGB-D images". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 564–571.

[157] Piotr Dollar, Zhuowen Tu, and Serge Belongie. "Supervised learning of edges and object boundaries". In: *Proceedings of the International conference on computer vision and pattern recognition*. Vol. 2. IEEE. 2006, pp. 1964–1971.

[158] Piotr Dollár and C Lawrence Zitnick. "Fast edge detection using structured forests". In: *IEEE transactions on pattern analysis and machine intelligence* 37.8 (2014), pp. 1558–1570.

[159] Jyri Kivinen, Chris Williams, and Nicolas Heess. "Visual boundary prediction: A deep neural prediction network and quality dissection". In: *Artificial Intelligence and Statistics*. 2014, pp. 512–521.

[160] Yun Liu et al. "Richer convolutional features for edge detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 3000–3009.

[161] Jianzhong He et al. "Bi-Directional Cascade Network for Perceptual Edge Detection". In: *Proceedings of the IEEE Conference on computer vision and pattern recognition*. 2019, pp. 3828–3837.

[162] Dan Xu et al. "Learning deep structured multi-scale features using attention-gated crfs for contour prediction". In: *Advances in Neural Information Processing Systems*. 2017, pp. 3961–3970.

[163] Yupei Wang et al. "Deep crisp boundaries: From boundaries to higher-level tasks". In: *IEEE Transactions on Image Processing* 28.3 (2018), pp. 1285–1298.

[164] Ruoxi Deng et al. "Learning to predict crisp boundaries". In: *Proceedings of the European Conference on Computer Vision*. 2018, pp. 562–578.

[165] Nathan Silberman et al. "Indoor segmentation and support inference from rgbd images". In: *Proceedings of the European Conference on Computer Vision*. Springer. 2012, pp. 746–760.

[166] David A Mély et al. "A systematic comparison between visual cues for boundary detection". In: *Vision research* 120 (2016), pp. 93–107.

[167] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. "Deepedge: A multi-scale bifurcated deep network for top-down contour detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 4380–4389.

[168] Chen-Yu Lee et al. "Deeply-supervised nets". In: *Artificial Intelligence and Statistics*. 2015, pp. 562–570.

[169] Ali Shahrokni, François Fleuret, and Pascal Fua. "Classifier-based contour tracking for rigid and deformable objects". In: *British machine vision conference*. CONF. 2005.

[170] Mukta Prasad et al. “Learning class-specific edges for object detection and segmentation”. In: *Computer Vision, Graphics and Image Processing*. Springer, 2006, pp. 94–105.

[171] Bharath Hariharan et al. “Semantic contours from inverse detectors”. In: *2011 International Conference on Computer Vision*. IEEE. 2011, pp. 991–998.

[172] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. “High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 504–512.

[173] Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017), pp. 834–848.

[174] Zhiding Yu et al. “Casenet: Deep category-aware semantic edge detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5964–5973.

[175] Yun Liu et al. “Semantic edge detection with diverse deep supervision”. In: *arXiv preprint arXiv:1804.02864* (2018).

[176] David Acuna, Amlan Kar, and Sanja Fidler. “Devil is in the edges: Learning semantic boundaries from noisy annotations”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 11075–11083.

[177] Kaiming He et al. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[178] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. “Dilated residual networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 472–480.

[179] Fisher Yu and Vladlen Koltun. “Multi-scale context aggregation by dilated convolutions”. In: *arXiv preprint arXiv:1511.07122* (2015).

[180] Xiaolong Wang et al. “Non-local neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7794–7803.

[181] Yue Cao et al. “Gcnet: Non-local networks meet squeeze-excitation networks and beyond”. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2019, pp. 0–0.

[182] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[183] Han Zhang et al. “Self-attention generative adversarial networks”. In: (2019), pp. 7354–7363.

[184] Han Hu et al. “Relation networks for object detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3588–3597.

[185] Hang Zhang et al. “Context encoding for semantic segmentation”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2018, pp. 7151–7160.

[186] Yanwei Li et al. “Attention-guided unified network for panoptic segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7026–7035.

[187] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[188] Xiang Li, Xiaolin Hu, and Jian Yang. “Spatial group-wise enhance: Improving semantic feature learning in convolutional networks”. In: *arXiv preprint arXiv:1905.09646* (2019).

[189] Jongchan Park et al. “Bam: Bottleneck attention module”. In: *arXiv preprint arXiv:1807.06514* (2018).

[190] Sanghyun Woo et al. “Cbam: Convolutional block attention module”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 3–19.

[191] Shanghua Gao et al. “Res2net: A new multi-scale backbone architecture”. In: *IEEE transactions on pattern analysis and machine intelligence* (2019).

[192] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. “Dynamic routing between capsules”. In: *Advances in neural information processing systems*. 2017, pp. 3856–3866.

[193] Steve Lawrence et al. “Neural network classification and prior class probabilities”. In: *Neural networks: tricks of the trade*. Springer, 1998, pp. 299–313.

[194] Tsung-Yi Lin et al. “Feature pyramid networks for object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[195] Rosanne Liu et al. “An intriguing failing of convolutional neural networks and the coordconv solution”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 9605–9616.

[196] Eli Gibson et al. “Automatic multi-organ segmentation on abdominal CT with dense v-networks”. In: *IEEE transactions on medical imaging* 37.8 (2018), pp. 1822–1834.

[197] Özgün Çiçek et al. “3D U-Net: learning dense volumetric segmentation from sparse annotation”. In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2016, pp. 424–432.

[198] Alexey A Novikov et al. “Fully convolutional architectures for multiclass segmentation in chest radiographs”. In: *IEEE transactions on medical imaging* 37.8 (2018), pp. 1865–1876.

[199] Gavin C Cawley and Nicola LC Talbot. “On over-fitting in model selection and subsequent selection bias in performance evaluation”. In: *Journal of Machine Learning Research* 11.Jul (2010), pp. 2079–2107.

[200] Louisa Lam, Seong-Whan Lee, and Ching Y Suen. “Thinning methodologies-a comprehensive survey”. In: *IEEE Transactions on pattern analysis and machine intelligence* 14.9 (1992), pp. 869–885.

[201] C Lawrence Zitnick and Piotr Dollár. "Edge boxes: Locating object proposals from edges". In: *Proceedings of the European conference on computer vision*. Springer. 2014, pp. 391–405.

[202] Hao Chen et al. "DCAN: deep contour-aware networks for accurate gland segmentation". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2016, pp. 2487–2496.

[203] David Bau et al. "Network dissection: Quantifying interpretability of deep visual representations". In: *Proceedings of the IEEE Conference on computer vision and pattern recognition*. 2017, pp. 6541–6549.

[204] Jan Kukačka, Vladimir Golkov, and Daniel Cremers. "Regularization for deep learning: A taxonomy". In: *arXiv preprint arXiv:1710.10686* (2017).

[205] Rich Caruana, Shumeet Baluja, and Tom Mitchell. "Using the future to" sort out" the present: Rankprop and multitask learning for medical risk evaluation". In: *Advances in neural information processing systems*. 1996, pp. 959–965.

[206] Sebastian Ruder. "An overview of multi-task learning in deep neural networks". In: *arXiv preprint arXiv:1706.05098* (2017).

[207] Zhanpeng Zhang et al. "Facial landmark detection by deep multi-task learning". In: *Proceedings of the European conference on computer vision*. Springer. 2014, pp. 94–108.

[208] Abhinav Valada, Noha Radwan, and Wolfram Burgard. "Deep auxiliary learning for visual localization and odometry". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE. 2018, pp. 6939–6946.

[209] Muming Zhao et al. "Leveraging heterogeneous auxiliary tasks to assist crowd counting". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 12736–12745.

[210] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1125–1134.

[211] Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks". In: *arXiv preprint arXiv:1511.06434* (2015).

[212] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. "Spatial transformer networks". In: *Advances in neural information processing systems*. 2015, pp. 2017–2025.

[213] Andre Esteva et al. "A guide to deep learning in healthcare". In: *Nature medicine* 25.1 (2019), p. 24.

[214] Andre Esteva et al. "Dermatologist-level classification of skin cancer with deep neural networks". In: *Nature* 542.7639 (2017), p. 115.

[215] Nicolas Coudray et al. "Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning". In: *Nature medicine* 24.10 (2018), p. 1559.

[216] Jeffrey De Fauw et al. "Clinically applicable deep learning for diagnosis and referral in retinal disease". In: *Nature medicine* 24.9 (2018), p. 1342.

[217] Varun Gulshan et al. “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs”. In: *Jama* 316.22 (2016), pp. 2402–2410.

[218] Paras Lakhani and Baskaran Sundaram. “Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks”. In: *Radiology* 284.2 (2017), pp. 574–582.

[219] Hyunkwang Lee et al. “An explainable deep-learning algorithm for the detection of acute intracranial haemorrhage from small datasets”. In: *Nature Biomedical Engineering* 3.3 (2019), p. 173.

[220] Zizhao Zhang et al. “Pathologist-level interpretable whole-slide cancer diagnosis with deep learning”. In: *Nature Machine Intelligence* 1.5 (2019), p. 236.

[221] Andrew Janowczyk and Anant Madabhushi. “Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases”. In: *Journal of pathology informatics* 7 (2016).

[222] Thijs Kooi et al. “Large scale deep learning for computer aided detection of mammographic lesions”. In: *Medical image analysis* 35 (2017), pp. 303–312.

[223] Morgan P McBee et al. “Deep learning in radiology”. In: *Academic radiology* 25.11 (2018), pp. 1472–1480.

[224] Maciej A Mazurowski et al. “Deep learning in radiology: An overview of the concepts and a survey of the state of the art with focus on MRI”. In: *Journal of Magnetic Resonance Imaging* 49.4 (2019), pp. 939–954.

[225] Chiyuan Zhang et al. “Understanding deep learning requires rethinking generalization”. In: *arXiv preprint arXiv:1611.03530* (2016).

[226] Davide Castelvecchi. “Can we open the black box of AI?” In: *Nature News* 538.7623 (2016), p. 20.

[227] Richard Bellman. “Dynamic programming”. In: *Science* 153.3731 (1966), pp. 34–37.

[228] NJ Raine-Fenning et al. “The reliability of virtual organ computer-aided analysis (VOCAL) for the semiquantification of ovarian, endometrial and subendometrial perfusion”. In: *Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology* 22.6 (2003), pp. 633–639.

[229] A Welsh. “Quantification of power Doppler and the index ‘fractional moving blood volume’(FMBV)”. In: *Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology* 23.4 (2004), pp. 323–326.

[230] AW Welsh. “A caution regarding standardization of power Doppler to measure perfusion in placental tissue”. In: *Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology* 31.1 (2008), pp. 111–112.

[231] Jonathan M Rubin et al. “Fractional moving blood volume: estimation with power Doppler US.” In: *Radiology* 197.1 (1995), pp. 183–190.