Improving Automated Lung Ultrasound Interpretation with Self-Supervised Learning

by

Blake VanBerlo

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2025

© Blake VanBerlo 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Ghassan Hamarneh Professor Computing Science Simon Fraser University
Supervisors:	Jesse Hoey Professor David R. Cheriton School of Computer Science University of Waterloo
	Alexander Wong Professor Systems Design Engineering University of Waterloo
Internal Members:	Jeff Orchard Associate Professor David R. Cheriton School of Computer Science University of Waterloo
	Yuri Boykov Professor David R. Cheriton School of Computer Science University of Waterloo
Internal-External Member:	Sirisha Rambhatla Assistant Professor Management Science and Engineering University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This thesis consists of eight chapters. Five chapters are based on published articles that were co-authored. Blake VanBerlo was the first author and major contributor on each article. Below, we list each article and indicate each co-author's contributions.

- **Chapter 2:** Blake VanBerlo, Jesse Hoey, and Alexander Wong. A survey of the impact of self-supervised pretraining for diagnostic tasks in medical X-ray, CT, MRI, and ultrasound. *BMC Medical Imaging*, 24(1):79, 2024. Publisher: Springer.

  Only Sections 2.3 and 2.4 in this thesis are adapted from the publication.

  - *Blake VanBerlo:* Conceptualization, methodology, investigation, data curation, writing - original draft, writing - review & editing, visualization, project administration
  - *Jesse Hoey:* Conceptualization, writing - review & editing, supervision
  - *Alexander Wong:* Conceptualization, writing - review & editing, supervision

- **Chapter 3:** Blake VanBerlo, Brian Li, Jesse Hoey, and Alexander Wong. Self-supervised pre-training improves performance and inference efficiency in multiple lung ultrasound interpretation tasks. *IEEE Access*, 11:135696–135707, 2023. Publisher: IEEE.

  - *Blake VanBerlo:* Conceptualization, methodology, software, validation, investigation, data curation, writing - original draft, writing - review & editing, visualization, project administration, funding acquisition
  - *Brian Li:* Software, writing - review & editing
  - *Jesse Hoey:* Writing - review & editing, supervision
  - *Alexander Wong:* Writing - review & editing, supervision

- **Chapter 4:** Blake VanBerlo, Brian Li, Alexander Wong, Jesse Hoey, and Robert Arntfield. Exploring the utility of self-supervised pretraining strategies for the detection of absent lung sliding in M-mode lung ultrasound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3076–3085, 2023.

  - *Blake VanBerlo:* Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing - original draft, writing - review & editing, visualization, project administration

- *Brian Li:* Software, writing - review & editing
- *Alexander Wong:* Writing - review & editing, supervision
- *Jesse Hoey:* Writing - review & editing, supervision
- *Robert Arntfield:* Resources, data curation, writing - review & editing

- **Chapter 5:** Blake VanBerlo, Alexander Wong, Jesse Hoey, and Robert Arntfield. Intra-video positive pairs in self-supervised learning for ultrasound. *Frontiers in Imaging*, 3, 2024.
  - *Blake VanBerlo:* Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing - original draft, writing - review & editing, visualization, project administration, funding acquisition
  - *Alexander Wong:* Methodology, writing - review & editing, supervision
  - *Jesse Hoey:* Methodology, writing - review & editing, supervision
  - *Robert Arntfield:* Resources, data curation, writing - review & editing
- **Chapter 6:** Blake VanBerlo, Jesse Hoey, Alexander Wong, and Robert Arntfield. The efficacy of semantics-preserving transformations in self-supervised learning for medical ultrasound. *Bioengineering*, 12(8):855, 2025. Publisher: MDPI.
  - *Blake VanBerlo:* Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing - original draft, writing - review & editing, visualization, project administration, funding acquisition
  - *Jesse Hoey:* Conceptualization, writing - review & editing, supervision
  - *Alexander Wong:* Conceptualization, writing - review & editing, supervision
  - *Robert Arntfield:* Resources, data curation, writing - review & editing

Abstract

Lung ultrasound (LU) is an increasingly important point-of-care examination in acute healthcare settings. In addition to exhibiting comparable accuracy to conventional imaging modalities such as radiography or computed tomography, ultrasound provides safety from radiation and enhanced portability at a reduced cost. Despite its purported benefits, LU has yet to be widely adopted due to a lack of trained experts and education programs. In response, machine learning algorithms have surfaced to alleviate the skills gap by providing automated interpretation. However, the training of machine learning algorithms requires vast amounts of manually annotated examples — that is, images labelled by an expert for whether or not they exhibit particular findings. Given that there are few experts qualified to provide quality annotations, there is a need to explore alternative machine learning techniques for this unique problem.

Self-supervised learning (SSL) has emerged as a class of methods to train machine learning algorithms to extract salient features from data without relying on labels. These so-called "pretrained" algorithms constitute better starting points when unlabelled data is abundant but labelled examples are scarce or challenging to acquire, as is the case for LU. Therefore, the purpose of this thesis is to propose and evaluate novel techniques tailored to lung ultrasound that enhance performance on core interpretation tasks, such as detecting lung sliding and pleural effusion.

The thesis consists of four studies. The first study explores the efficacy of SSL techniques in brightness mode LU interpretation tasks and proposes a multi-task framework that reuses a single pretrained algorithm to efficiently interpret LU images. The results demonstrate that SSL improves performance on multiple classification tasks. In addition, experiments show that pretrained algorithms amplify generalizability to external healthcare centres. The second study introduces SSL methods to motion mode ultrasound and proposes data augmentation techniques specific to motion mode. We observe that pretrained algorithms achieve the greatest performance on local and external test data for the challenging task of lung sliding classification. The third study proposes a LU-specific technique for SSL that involves sampling pairs of images from the same ultrasound video that are temporally or spatially proximal to each other, based on the intuition that such pairs of images share similar content. The results indicate that, with appropriate parameter assignments, this sampling strategy improves performance of pretrained algorithms on multiple tasks. Lastly, the fourth study proposes ultrasound-specific data augmentation and image preprocessing methods for SSL. The results underscore the value of ultrasound-specific image preprocessing in SSL. A comprehensive evaluation finds that ultrasound-specific data

augmentation yields the best performance on a diagnostic task, and that techniques based on cropping attain top performance on object classification tasks.

Overall, the findings of this thesis demonstrate that SSL improves the performance of machine learning algorithms for LU, especially on LU images originating from external healthcare centres. Novel SSL methods for LU are established as a key ingredient for producing algorithms that are effective for multiple LU interpretation tasks.

Acknowledgements

First, I would like to thank my co-supervisors, Dr. Jesse Hoey and Dr. Alexander Wong. Jesse has been an impactful mentor who has guided me since I started my graduate studies. He has always believed in my abilities and frequently took the initiative to expose me to new opportunities that have enriched my professional development. When I started my research, Alex agreed to take me on as a student, lending his domain expertise to this project. He has continuously provided me with invaluable feedback and professional mentorship. Both Jesse and Alex have been stellar, forthright, and encouraging supervisors. They also supported my pursuit of parallel interests in teaching and entrepreneurship. I am lucky to have found supervisors who challenged and emboldened me.

Next, I would like to acknowledge the remaining members of my examining committee: Professors Ghassan Hamarneh, Jeff Orchard, Yuri Boykov, and Sirisha Rambhatla. I am thankful for their time and effort spent reviewing my work, attending my seminars, and providing insightful criticism.

I would also like to thank Dr. Robert Arntfield, an incredible mentor, collaborator, and friend. From his clinical expertise, to the data required to conduct lung ultrasound machine learning experiments at scale, Rob's contributions were pivotal to this thesis. More importantly, his mentorship has had a tremendous impact on me, facilitating professional and personal growth for which I am sincerely appreciative.

Considerable financial assistance and computational resources were instrumental to this thesis. I am grateful to the Natural Sciences and Engineering Research Council of Canada, who awarded me a Vanier Scholarship to pursue this research. The award gave me the freedom to work on a unique problem that I believe is important. Additionally, I would like to acknowledge Compute Ontario, BC DRI Group, and the Digital Research Alliance of Canada, who provided the computational infrastructure required to conduct the myriad machine learning experiments in this thesis.

Next, I would like to thank my family and friends for their love, encouragement, and patience as I worked on this research. To my parents-in-law, David and Isabel, I am thankful for your continuous belief in my endeavours. To my brothers (and best friends), Brock and Bennett, thank you for unwaveringly being in my corner. To my father, Mark, and his fiancée, Mickey — thank you for being there for me, wherever and whenever. I am also deeply grateful to my mother, Lori, for nurturing my curiosity and work ethic throughout childhood, which I believe was essential for me to reach this milestone.

Finally, I am profoundly grateful to my wife, Olivia. She has been a constant source of motivation, support, love, laughter, and happiness throughout my graduate studies. Olivia brings out the best in me and inspires me to push my capabilities.

Dedication

I dedicate this work to my wife, Olivia, and to my father, Mark.

Your support throughout this journey has meant more to me than I can ever articulate.

Table of Contents

Examining Committee Membership ii

Author's Declaration iii

Statement of Contributions iv

Abstract vi

Acknowledgements viii

Dedication ix

List of Figures xvii

List of Tables xix

List of Abbreviations xxii

1 Introduction 1

1.1 Motivation 1

1.2 Problem Overview 3

1.3 Contributions and Outline 5

2 Background **7**
2.1 Introduction to Medical Ultrasound . . . 7
2.2 Lung Ultrasound Tasks . . . 10
2.2.1 Overview of Lung Ultrasound . . . 10
2.2.2 Lung Ultrasound Machine Learning Tasks . . . 12
2.3 Self-Supervised Learning . . . 17
2.3.1 Motivation for Self-Supervised Learning . . . 17
2.3.2 Self-Supervised Learning Preliminaries . . . 19
2.3.3 Taxonomy of Self-Supervised Learning . . . 21
2.3.4 Theoretical Support . . . 26
2.4 Self-Supervised Learning in Ultrasound . . . 26
2.4.1 Literature Search Methodology . . . 26
2.4.2 Applications in Ultrasound Imaging . . . 27
2.4.3 Key Literature Survey Findings . . . 33

3 Self-Supervised Pretraining for Multiple Tasks in B-Mode Lung Ultrasound **39**
3.1 Introduction . . . 39
3.2 Background . . . 42
3.2.1 Joint Embedding Self-Supervised Learning . . . 42
3.2.2 Multi-Task Medical Image Interpretation . . . 43
3.3 Methods . . . 43
3.3.1 Lung Ultrasound Classification Tasks . . . 43
3.3.2 Datasets . . . 43
3.3.3 Self-Supervised Pretraining . . . 45
3.3.4 Evaluation Protocol . . . 46
3.4 Results . . . 48
3.4.1 Linear Classifier Evaluation (LC) . . . 48

3.4.2 Fine-tuning Evaluation (FT) . . . 49
3.4.3 Nonlinear Classification (NC) . . . 50
3.4.4 Public Dataset Evaluation . . . 51
3.4.5 Label Efficiency . . . 51
3.4.6 Qualitative Evaluation of Representations . . . 51
3.4.7 Inference Efficiency . . . 52
3.5 Discussion . . . 54

4 Self-Supervised Pretraining for Lung Sliding Classification in M-Mode Lung Ultrasound 56
4.1 Introduction . . . 56
4.2 Background . . . 58
4.2.1 Lung Sliding Classification . . . 59
4.2.2 Pairwise Relationship in Self-Supervised Learning . . . 60
4.3 Methods . . . 60
4.3.1 Dataset . . . 60
4.3.2 Self-Supervised Learning . . . 62
4.3.3 Evaluation Protocol . . . 64
4.4 Results . . . 66
4.4.1 Data Augmentation . . . 66
4.4.2 Comparison with Supervised Baselines . . . 67
4.4.3 Label Efficiency . . . 68
4.4.4 Evaluation on External Datasets . . . 69
4.4.5 Explainability . . . 70
4.5 Discussion . . . 71

5 Intra-video Positive Pairs in Self-Supervised Learning for Ultrasound 74
5.1 Introduction . . . 74
5.2 Related Works . . . 77
5.3 Methods . . . 78
5.3.1 Joint Embedding Methods with Intra-Video Positive Pairs . . . 78
5.3.2 Ultrasound Classification Tasks: . . . 81
5.3.3 COVID-19 Classification (COVID) . . . 81
5.4 Results . . . 83
5.4.1 Training Protocols . . . 83
5.4.2 Performance . . . 85
5.4.3 Label Efficiency . . . 87
5.5 Discussion . . . 91
5.5.1 Guidelines for Machine Learning Practitioners . . . 91
5.5.2 Limitations . . . 93
5.6 Conclusion . . . 94

6 The Efficacy of Semantics-Preserving Transformations in Self-Supervised Learning for Lung Ultrasound 96
6.1 Introduction . . . 96
6.2 Background . . . 98
6.2.1 Data Augmentation in Self-Supervised Learning . . . 98
6.2.2 Data Augmentation for Self-Supervision in Ultrasound . . . 99
6.3 Methods . . . 99
6.3.1 Datasets and Tasks . . . 99
6.3.2 Semantics-Preserving Preprocessing . . . 101
6.3.3 Ultrasound-specific Data Augmentation . . . 101
6.3.4 Discovering Semantically Contributory Transformations . . . 106
6.3.5 Training Protocols . . . 107

6.4 Experiments and Results . . . 107
6.4.1 Transformation Leave-one-out Analysis . . . 107
6.4.2 Object Classification Task Evaluation . . . 108
6.4.3 Diagnostic Classification Task Evaluation . . . 113
6.4.4 Object Detection Task Evaluation . . . 116
6.4.5 Label Efficiency Assessment . . . 117
6.4.6 Impact of Semantics-Preserving Preprocessing . . . 117
6.4.7 Impact of the Cropping Transformation in Object Classification Tasks 118
6.5 Conclusion . . . 121

7 Conclusions and Future Work **123**
7.1 Conclusions . . . 123
7.1.1 Primary Results . . . 123
7.1.2 Recommendations . . . 125
7.2 Directions for Future Work . . . 126
7.2.1 Masked Image Modelling for Lung Ultrasound . . . 126
7.2.2 Video-Level Self-Supervised Learning . . . 128
7.2.3 Reusable Backbones for Efficient Multi-Task Inference . . . 129
7.2.4 Does Self-Supervised Learning Improve Generalizability? . . . 131
7.3 Closing Words . . . 132

References **134**

APPENDICES **158**

A Private Dataset Access **159**

B Additional Classification Metrics for Chapter 3 **160**

C Self-supervised Learning Objectives with Sample Weights **162**
C.1 Weighted SimCLR Objective . . . 162
C.2 Weighted VICReg Objective . . . 162
C.3 Weighted Barlow Twins Objective . . . 163

D Intra-Video Positive Pair Results **165**

E Batch Size Robustness for Intra-Video Positive Pairs **169**

F Data Augmentation Details **171**
F.1 StandardAug Transformations . . . 171
F.1.1 Crop and Resize (B00) . . . 171
F.1.2 Horizontal Reflection (B01) . . . 172
F.1.3 Color Jitter (B02) . . . 172
F.1.4 Conversion to Grayscale (B03) . . . 172
F.1.5 Gaussian Blur (B04) . . . 172
F.1.6 Solarization (B05) . . . 172
F.2 Ultrasound-Specific Transformations . . . 172
F.2.1 Probe Type Change (U00) . . . 173
F.2.2 Convexity Change (U01) . . . 174
F.2.3 Wavelet Denoising (U02) . . . 176
F.2.4 Contrast-Limited Adaptive Histogram Equalization (U03) . . . 177
F.2.5 Gamma Correction (U04) . . . 177
F.2.6 Brightness and Contrast Change (U05) . . . 177
F.2.7 Depth Change Simulation (U06) . . . 177
F.2.8 Speckle Noise Simulation (U07) . . . 178
F.2.9 Gaussian Noise Simulation (U08) . . . 178
F.2.10 Salt and Pepper Noise Simulation (U09) . . . 178
F.2.11 Horizontal Reflection (U10) . . . 178
F.2.12 Rotation and Shift (U11) . . . 178

G Pleural Line Object Detection Training 179

H Data Augmentation Leave-one-out Analysis Statistical Testing 181

I Additional Positive Pair Examples 184

J Data Augmentation Comparisons with ResNet18 Backbone 187

K Data Augmentation Label Efficiency Statistical Testing 191

Glossary 193

List of Figures

2.1 Examples of B-Mode ultrasound images captured by different transducer types. . . . 8
2.2 Composition of M-mode images . . . 9
2.3 Key structures visible in each lung ultrasound view . . . 11
2.4 Examples of ultrasound images from each class . . . 13
2.5 Hierarchical relationship between lung ultrasound interpretation tasks . . . 14
2.6 Overview of self-supervised learning . . . 18
2.7 Examples of generative pretext tasks . . . 22
2.8 Examples of predictive pretext tasks . . . 22
2.9 The forward pass for a positive pair in joint embedding SSL . . . 24
2.10 Distribution of papers in the literature survey by imaging modality and year of publication . . . 28

3.1 An overview of the methods in Chapter 3 . . . 40
3.2 The forward pass for a batch of images in joint embedding self-supervised learning . . . 42
3.3 Examples of ultrasound images from each class for the downstream tasks in Chapter 3 . . . 44
3.4 Examples of positive pairs of B-mode images . . . 47
3.5 Test set AUC for fully supervised and SimCLR-pretrained models . . . 52
3.6 Test set t-SNE projections for ImageNet-pretrained and SimCLR-pretrained models . . . 53

4.1 Overview of the methods in Chapter 4 . . . 58
4.2 Dataset splits in Chapter 4 . . . 61
4.3 Positive pairs of M-mode images . . . 63
4.4 M-mode images after applying data augmentation . . . 64
4.5 Label efficiency for fully supervised and Barlow Twins-pretrained models . . . 69
4.6 Saliency maps for fully supervised and pretrained lung sliding classifiers . . . 71

5.1 Overview of methods in Chapter 5 . . . 76
5.2 Intra-video positive pairs for B-mode and M-mode . . . 80
5.3 Test set performance on POCOVID-Net . . . 86
5.4 Test set performance on ParenchymalLUS . . . 88
5.5 Label efficiency experiment for IVPP . . . 91

6.1 Overview of the methods in Chapter 6 . . . 97
6.2 Semantics-preserving ultrasound preprocessing . . . 103
6.3 Examples of data augmentation transformations from the AugUS-O pipeline applied to the same lung ultrasound image. . . . 105
6.4 Examples of positive pairs under various data augmentation pipelines . . . 109
6.5 LUSData test set t-SNE projections . . . 112
6.6 Label efficiency experiment for models pretrained using various data augmentation pipelines . . . 118
6.7 Examples of random crop and resize applied to ultrasound images . . . 120
6.8 Test set performance for models pretrained using a range of minimum crop sizes . . . 121

7.1 A depiction of a multi-task LU interpretation system . . . 130

F.1 Locations of named vertices on ultrasound field of view . . . 173

I.1 Positive pairs produced using the StandardAug pipeline . . . 185
I.2 Positive pairs produced using the AugUS-O pipeline . . . 185
I.3 Positive pairs produced using the AugUS-D pipeline . . . 186

List of Tables

2.1 Details on public ultrasound datasets 38

3.1 A summary of the lung ultrasound tasks addressed in Chapter 3 44
3.2 Breakdown of the datasets used in Chapter 3 45
3.3 Test set performance for lung ultrasound downstream classification tasks with and without self-supervised pretraining 48
3.4 Test AUC on the COVIDx-US dataset 50

4.1 Class distribution for datasets in in Chapter 4 61
4.2 Validation set AUC for linear M-mode classifiers 67
4.3 Downstream task performance for lung sliding classification with random backbone initialization 68
4.4 Downstream task performance for lung sliding classification with ImageNet backbone initialization 68
4.5 Downstream classification performance of models pretrained using Barlow Net using labelled and unlabelled datasets. 69
4.6 External test set lung sliding classification performance across external datasets 70

5.1 Breakdown of the ParenchymalLUS dataset 82
5.2 Comparison of IVPP and other ultrasound-specific contrastive learning methods 89
5.3 Label efficiency experiment for IVPP 92

6.1 Characteristics of the COVIDx-US dataset 100

6.2 Label distribution of the LUSData dataset . . . 101
6.3 Characteristics of the COVIDx-US dataset . . . 102
6.4 The sequence of transformations in the StandardAug data augmentation pipeline . . . 103
6.5 The sequence of transformations in the ultrasound-specific augmentation pipeline . . . 106
6.6 Leave-one-out analysis for data augmentation transformations . . . 110
6.7 LUSData test set performance using different data augmentation pipelines 111
6.8 External test set metrics under various data augmentation pipelines . . . 114
6.9 Test set performance for the `COVID` task . . . 115
6.10 LUSData local test set AP@50 for the `PL` task . . . 116
6.11 Test set performance with and without semantics-preserving preprocessing 119

B.1 Additional local test set metrics for experiments in Chapter 3 . . . 161
B.2 Additional external test set metrics for experiments in Chapter 3 . . . 161

D.1 Mean cross-validation accuracy for pretrained classifiers on the POCUSBorn dataset . . . 166
D.2 ParenchymalLUS test set performance for linear classifiers . . . 167
D.3 ParenchymalLUS test set performance for fine-tuned classifiers . . . 168

E.1 Comparison of downstream task performance under various pretraining batch sizes . . . 170

G.1 MobileNetV3Small feature map dimensions . . . 180

H.1 Friedman test statistics for leave-one-out analysis for data augmentation . 182
H.2 Wilcoxon Sign-Rank test statistics for leave-one-out analysis for data augmentation . . . 183

J.1 Test set performance for pretrained ResNet18 models . . . 189
J.2 External test set performance for pretrained ResNet18 models . . . 190

K.1 Wilcoxon Sign-Rank test statistics for label efficiency experiments comparing data augmentation pipelines for the AB task 192
K.2 Wilcoxon Sign-Rank test statistics for label efficiency experiments comparing data augmentation pipelines for the PE task 192

List of Abbreviations

ANOVA analysis of variance 89

AUC area under the receiver operating characteristic curve 47

BYOL Bootstrap Your Own Latent 25

C&R crop and resize 115

CNN convolutional neural network 3

CT computed tomography 2

DICOM Digital Imaging and Communications in Medicine 30

DNN deep neural network 3

FOV field of view 101

FT fine-tuning 46

IVPP Intra-Video Positive Pairs 5

LC linear classification 46

LU lung ultrasound 1

MAE masked Autoencoder 22

MIM masked image modelling 22

MLP multilayer perceptron 20

MRI magnetic resonance imaging 2

NC nonlinear classification 46

POCUS point-of-care ultrasound 1

PTX pneumothorax 8

SSD Single Shot Detector 107

SSL self-supervised learning 1

USCL Ultrasound Contrastive Learning 29

Chapter 1

Introduction

Lung ultrasound (LU) is a technique for imaging the lungs to obtain quick answers to pertinent diagnostic questions.[1] Despite the growing body of literature supporting its use, it has yet to be widely adopted due to a lack of training and difficulty in interpretation. Machine learning strategies for LU interpretation are expanding, however they are limited by a lack of training data. This thesis focuses on the improvement of automated LU interpretation using self-supervised learning (SSL) techniques.

The introductory chapter of this thesis summarizes the efforts described herein. Section 1.1 elucidates the motivation for improving automated interpretation of LU. Following this, Section 1.2 provides a clear overview of some of the obstacles to developing robust automated interpretation solutions for LU interpretation. Lastly, Section 1.3 outlines the main contributions of this work and outlines the organization of this thesis.

1.1 Motivation

Point-of-care ultrasound (POCUS) is a rapidly evolving type of medical imaging that is revolutionizing diagnostics in acute healthcare [143]. This eponymous imaging modality facilitates image acquisition and interpretation at the site of the patient. It is made possible by ultrasound imaging, wherein which a probe emits sound waves and uses those that are reflected back to compose an image of underlying anatomical structures.

[1] We abbreviate "lung ultrasound" as "LU" instead of the commonly used — but erroneous — "LUS", to respect the wishes of Daniel Lichtenstein, one of the core academic contributors to lung ultrasound [94].

Several attributes of POCUS distinguish it as a useful and versatile diagnostic tool. First, it is highly portable – ultrasound devices can be transported directly to a patient's bedside in a hospital setting. Other imaging modalities – such as X-ray, computed tomography (CT) and magnetic resonance imaging (MRI) – often require patients to be transported to the imaging device. With the advent of handheld ultrasound devices, diagnostic imaging can be conducted anywhere that a stethoscope can be taken. As a result, POCUS can be employed in a plurality of environments, such as pre-hospital settings [20], rural and remote clinics [178], military conflicts [40], and outer space [134]. Second, POCUS is safe for patients. Unlike X-ray and CT, ultrasound does not emit ionizing radiation. Lastly, POCUS is at least as accurate as other imaging modalities for a variety of critical findings. When considering its diagnostic accuracy in the context of its unmatched portability, low cost, and safety, POCUS is a desirable choice in several settings. Despite mounting evidence for its efficacy, there are barriers to the widespread adoption of POCUS, including reduced availability of training and lack of access to devices [21, 54, 58].

Lung ultrasound (LU) is a type of POCUS examination that is performed in a variety of care settings to rapidly narrow down differential diagnoses for patients in respiratory distress. There are a variety of LU protocols, but they all involve assessing the lungs and the pleura for a variety of key structures and artifacts. Examples of conditions that LU can help to identify or rule out include pneumonia, pneumothorax, tuberculosis, and heart failure. In addition to the enhanced portability, safety, and affordability conferred by POCUS, LU has exhibited diagnostic accuracy for a variety of respiratory conditions that is comparable to traditional modalities, such as chest radiography [2, 91, 116, 169] and computed tomography [34, 116, 117, 164]. LU can also be interpreted at the time of acquisition, preventing the need to transport acutely ill patients to X-ray or CT scanners and await interpretation [179].

Compared to more commonplace applications of POCUS, such as echocardiography, LU is a nascent technique. In the past, it was widely believed that ultrasound imaging was ineffectual for assessing the lungs because sound waves scatter upon entering aerated mediums [51]. However, clinical scholars discovered that these noise-laden thoracic ultrasound views were not bereft of signals. Several studies established the diagnostic value in identifying and characterizing artifacts generated by the unique acoustic phenomena that ensures when ultrasonic waves interact with normal and diseased lung tissue [92, 143]. Other signs involving thoracic structures and contents that are external to the lung have also been described [95].

Although there exists a strong body of literature demonstrating the diagnostic utility of lung ultrasound, it has yet to be widely adopted by clinicians. Major barriers include a lack of training programs, experts, and image archival systems [64, 75, 136]. Studies have shown

that healthcare providers trained to perform LU do not exhibit high inter-rater agreement for key findings [57, 128]. Moreover, some findings are more difficult to appreciate than others. For example, nurses who participated in a LU training program were frequently able to correctly identify a B-line profile but struggled to distinguish lung sliding [114]. To address the vacuum of experts and training programs, automated interpretation programs have been developed that identify key signs in LU. A plenitude of deep learning solutions has been described for classifying or localizing findings in LU [5, 6, 26, 42, 78, 106, 146, 158]. Automated solutions can assist in the education of novel POCUS users, provide a secondary opinion for competent sonographers, and make LU accessible to a broader set of allied healthcare practitioners. Most critically, automation can bring real-time respiratory diagnostics to patients in diverse categories of environments. In doing so, it can provide immediate access to information that could influence the clinical trajectory that could otherwise take hours. The expeditious availability of such information is especially valuable in acute scenarios such as trauma settings.

1.2 Problem Overview

Despite the rising amount of evidence supporting deep learning approaches to lung ultrasound image interpretation, there are critical barriers inhibiting their development. There are two primary obstacles that researchers and developers face: data availability and LU expertise.

As with any deep learning problem, developers require an abundance of labelled examples that can be used to train a deep neural network (DNN). Modern DNN architectures such as convolutional neural networks (CNNs) often require thousands of examples per class at minimum for classification tasks. Vision transformers often require even more examples, since they lack the inductive bias inherent to CNNs that facilitates local feature aggregation [144]. The number of labelled examples required can grow depending on several factors, such as the complexity of the task, the difficulty of the task, and burden of label noise. Access to labelled data is a known barrier in machine learning for POCUS [102]. Large-scale LU datasets are particularly difficult to obtain for multiple reasons. First, LU has a brief history, having not been established nearly as long as other medical imaging modalities or POCUS examinations. Second, POCUS exams are not routinely documented in picture archiving and communication systems [63,84]. Similar to physical examinations, information procurement and interpretation occurs simultaneously. As a result, healthcare institutions may not have any retrospectively collected examinations from which to source LU datasets for machine learning. Note that although machine learning solutions for LU

largely operate on still images, collections of videos do not constitute effective LU datasets for machine learning because images in each video strongly resemble each other, particularly when the ultrasound probe remains in a fixed position over the duration of scanning. Furthermore, multiple videos are typically collected from the same patient in the course of an examination and are therefore correlated. Thus, datasets comprising a couple hundred POCUS videos are not large collections of independent and identically distributed images.

In addition to limited data availability, supervised learning problems require labels associated with each example. Compared to most computer vision application domains, labelling expertise in medical imaging can be expensive. Healthcare providers qualified to provide labelling services (especially physicians) demand high compensation due to their specialized skill sets and clinical demands on their time. Costs are exceptionally high when the set of experts is small. Unfortunately, there are very few trained experts in LU [75]. Moreover, LU can be particularly difficult to label. Low inter-rater agreement for findings in LU [57, 128] underline the need for expertise to establish trustworthy ground truth. Occasionally, retrospectively collected examples are available through archiving systems. However, it is not commonplace for dedicated imaging reports to accompany POCUS examinations [63,84]. As such, when LU videos are abundant – either through prospective collection or in a retrospectively sourced archive – labelling presents a challenge.

In cases where unannotated retrospective data is abundant, labelling efforts may be prohibitive due to the cost and availability of LU expertise. Recent developments in deep computer vision have studied the utility of unsupervised pretraining on large unlabelled datasets as a means to produce DNNs that extract semantic features, which may be adapted for particular tasks via transfer learning. SSL has emerged as a leading paradigm in representation learning for images. Broadly speaking SSL refers to an unsupervised learning strategy in which DNNs are trained to predict withheld or unknown attributes of examples themselves. Studies have shown that pretraining with SSL and fine-tuning using a small subset of examples can lead to comparable results when training from scratch using an entire labelled dataset [12,29,59,180]. It confers significant value in medical imaging tasks in which labels are only available for a subset of examples or are expensive to obtain. This thesis studies the utility of SSL techniques for LU interpretation tasks. It proposes and compares SSL mechanisms that are tailored for LU. The results largely indicate that SSL pretraining is useful for multiple interpretation tasks in LU.

1.3 Contributions and Outline

The main objective of this thesis is to enhance the efficacy of automated LU interpretation. To this end, we investigate and propose new SSL methods for LU. As discussed in prior sections, SSL is a befitting approach for developing machine learning algorithms for LU because it reduces the burden of obtaining labelled data. This thesis establishes that SSL tailored to LU materially improves the performance and generalizability of machine learning algorithms for automated LU interpretation. Novel domain-specific methods tailored to SSL are proposed and comprehensively evaluated. Guidance for machine learning practitioners is synthesized across chapters, based on the experimental results. Below, we summarize the contributions of each chapter.

Chapter 1 (the current chapter) introduces the reader to the problems addressed in this thesis. The importance of developing automated solutions for LU interpretation is discussed, along with accompanying challenges.

Chapter 2 provides the reader with the necessary background knowledge to navigate the subsequent chapters in the thesis. It begins by reviewing the basics of medical ultrasound and describes the LU interpretation tasks that are referenced throughout the thesis. SSL is then defined, and some technical preliminaries are introduced. What follows is an account of the evidence in the literature regarding the impact SSL in machine learning for medical ultrasound. The chapter finishes with an extensive description of the LU interpretation tasks that are utilized for the experiments in the subsequent chapters.

Chapter 3 puts forward a multi-task interpretation framework that features a single self-supervised backbone whose features are inputted to hierarchically related lightweight LU classifiers. The experiments reveal that a single pretrained backbone improves performance on multiple B-mode LU classification tasks. It also finds that pretrained models enjoy greater label efficiency and improved performance on external test data.

Chapter 4 details an SSL approach for improving performance on a critical task in M-mode LU — lung sliding classification. It proposes and evaluates data augmentation methods for SSL specifically with M-mode LU. The experiments also investigate the effect of pretraining data volume, label efficiency, saliency maps for pretrained models, and generalizability to external datasets.

Chapter 5 introduces a novel positive pair sampling method for SSL that is tailored to ultrasound. Intra-Video Positive Pairs (IVPP) samples semantically related pairs of images from the same B-mode video that are temporally close to each other. The approach is extended to M-mode ultrasound by sampling spatially vicinal M-mode images from the same B-mode video. A sample weighting scheme is also introduced that weighs positive

pairs according to their degree of proximity. The sample weight strategy may be readily incorporated into the objective functions of any joint embedding SSL method, and its application is not restricted to LU. The experiments assess the effect of the positive pair sampling strategy and sample weighting scheme, for a variety of SSL pretraining objectives.

Chapter 6 introduces data augmentation and preprocessing techniques for SSL that are tailored to LU. A novel data augmentation strategy is designed to preserve semantic content in LU images. The experiments uncover the most contributory transformations across multiple data augmentation pipelines according to downstream classification performance across multiple B-mode LU tasks. We find that the top-performing data augmentation strategy differs according to the type of LU classification task. Lastly, ultrasound-specific image preprocessing in SSL is found to improve performance across multiple tasks.

Finally, Chapter 7 concludes the thesis. An aggregation of results from each study is presented and distilled into guidance for machine learning practitioners. Several directions for future work are provided.

Chapter 2

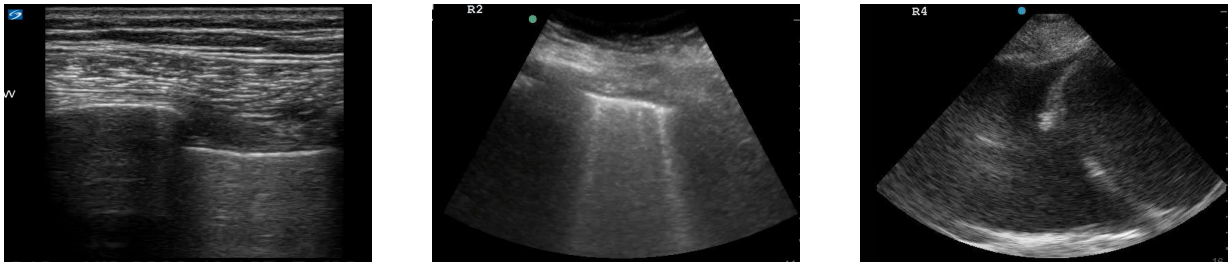
Background

2.1 Introduction to Medical Ultrasound

Ultrasound is diagnostic imaging modality that uses the transmission and reflection of high-frequency sound waves to obtain information about a patient's underlying anatomy. A piezoelectric crystal on the ultrasound device emits sound waves, which are reflected, absorbed, and transmitted by tissues. Different types of tissues exhibit varying resistance to the passage of sound waves (i.e., different *acoustic impedance*). When sound waves pass from one type of tissue to another with a different acoustic impedance, some of the waves are reflected. A greater difference in acoustic impedance between adjacent tissues leads to greater amplitude of reflected waves. Waves that are reflected back to the ultrasound probe are received by a group of transducers. The amplitudes of the received waves are converted to an electrical signal, which is used to generate the grayscale image [143]. We begin by describing two main modes of ultrasound imaging: B-mode and M-mode.

B-mode Ultrasound

The most commonly employed ultrasound mode is *B-mode* (i.e., *brightness mode*), which consists of a video stream of two-dimensional grayscale images. The image is oriented such that regions at the top are the closest to the transducer, and signals are plotted in positions that correspond to the time taken for the reflected waves to return to the probe. The pixel intensity of structures visible in the images is dependent on the *echogenicity* (i.e., tendency to reflect sound waves) of the tissues with respect to surrounding structures [143]. Black regions correspond to *anechoic* (non-reflective) regions that typically contain fluid, such



(a) Linear transducer. (b) Curvilinear transducer. (c) Phased array transducer.

Figure 2.1: Examples of B-Mode ultrasound images captured by different transducer types.

as cysts and vasculature. *Hypoechoic* structures reflect some sound waves, but less than surrounding tissues, whereas *isoechoic* structures reflect waves similarly to surrounding tissues (e.g., muscle). Such regions appear as varying shades of grey. *Hyperechoic* structures reflect most sound waves and appear white (e.g., diaphragm, bones) [143].

There are three basic types of transducers: linear, curvilinear, and phased array. The type of transducer determines the shape of the beam (i.e., field of view) in B-mode images. The intensity of any pixels surrounding the field of view is set to 0. Figure 2.1 provides examples of types of B-mode images.

M-Mode Ultrasound

Generally applied to visualize the motion of structures over time, *M-mode* (i.e., *motion mode*) ultrasound consists of a two-dimensional image in which time constitutes the horizontal axis. To capture M-mode feed, the sonographer designates a line within a B-Mode image. The device emits a beam along this axis, which then forms the vertical of the M-Mode image, and the operator can visualize this linearly arranged set of pixels over time [143]. Figure 2.2 portrays the process by which M-Mode image capture is set up.

M-Mode is useful for assessing artefacts produced by the motion of organs and tissues. An example that will be studied in this research is the distinction between the presence and absence of the *lung sliding* finding in lung ultrasound. Visible on B-Mode as the shimmering of the *pleural line* (see Figure 2.3a), lung sliding is 100% specific for pneumothorax (PTX) at the site of the ultrasound probe [95], ruling PTX out of a differential diagnosis. On M-Mode LU, the *seashore sign* [92] and *barcode sign* [97] are two visually distinct patterns that signal the presence and absence of lung sliding, respectively (see Figure 2.4c). M-

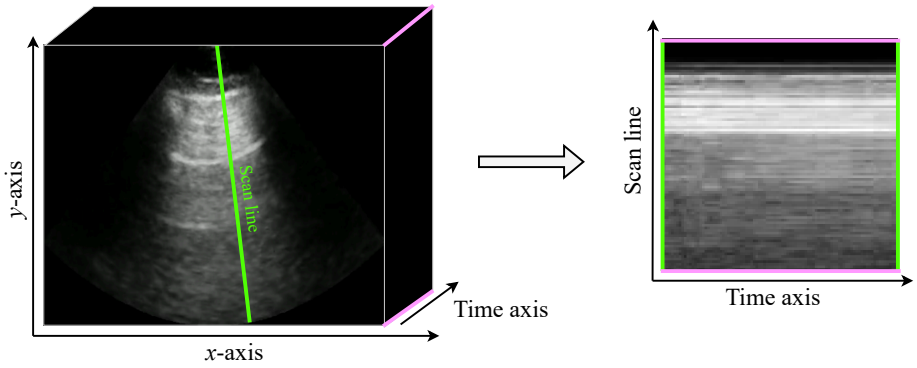


Figure 2.2: M-Mode images depict a one-dimensional component of an ultrasound beam over time. The clinician designates the scan line on the B-Mode image (green) that becomes the vertical axis of the M-Mode image. The time dimension in the B-mode image (pink) becomes the horizontal axis in the M-mode image.

Mode is also widely used in *echocardiography* (i.e., ultrasound of the heart) for determining the size and motion of the cardiac chambers and valves [143].

Ultrasound Device Parameters

Ultrasound devices are equipped with a plethora of tools to modify image acquisition according to the type and purpose of examination. There are two main parameters that the sonographer can modify to adjust the image. First, the sonographer can adjust the *depth* of the ultrasound beam to sharpen the image at the structures of interest. The depth determines the distance from the probe to the focal zone, where lateral resolution is greatest [143]. The gain of the image can also be adjusted, which affects the brightness of non-anechoic structures. Increasing the gain results in brighter images overall [143]. Judicious tuning of these parameters is important for producing discernible images. Several modern devices are equipped with preset assignments for acquisition parameters that are optimized for examinations of various anatomical locations.

Noise in Ultrasound Imaging

Multiple characteristics of ultrasound imaging may render it difficult to interpret. Ultrasound images are often low resolution, distort the geometrical form of anatomical struc-

tures, and are subject to noise degradation. There are characteristic noise distributions inherent to ultrasound imaging, which can reduce the interpretive quality of images, thereby blocking appreciation of finer details [112].

Speckle noise is the predominant type of multiplicative noise in ultrasound imaging, and it is caused by constructive and destructive interference of the acoustic waves emitted by and reflected back to the device [56, 61, 112]. Multiple studies have suggested methods for the removal of speckle noise, such as nonlinear filters [112], Laplace mixture distributions [50], and deep convolutional neural networks [81]. Gaussian noise is also present in ultrasound imaging, typically introduced by thermal and electronic fluctuations in the device [61, 81]. Salt and pepper noise is a pixel-wise phenomenon in which individual pixel intensities are erroneously set to distinct fixed values, which are generally black or white [61]. Lastly, Poisson noise is a rare form of noise caused by fluctuations in electric charge [61].

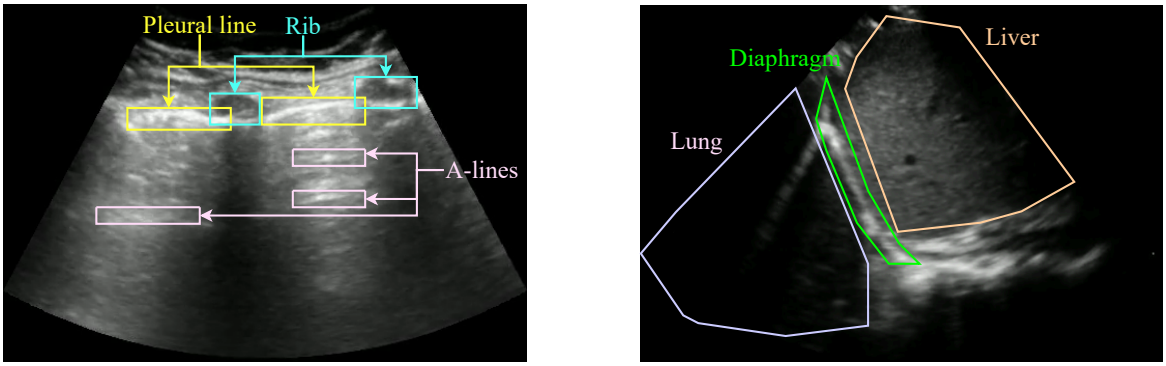
2.2 Lung Ultrasound Tasks

In this section, we describe the LU interpretation tasks that are used throughout this thesis for experimental evaluation of machine learning algorithms. It is intended to serve as a reference that the reader may consult for clarification on the machine learning problems on which the methods are evaluated. We begin with a primer on lung ultrasound, then define interpretation tasks.

2.2.1 Overview of Lung Ultrasound

Lung ultrasound refers to the use of POCUS to examine the lungs and the pleura (i.e., lining of the lung). A single examination consists of the collection and assessment of multiple videos at various sites on the thorax. Typically, the patient is in a supine position. Two major types of views are collected: *parenchymal views* and *pleural views* (see Figure 2.4a). Parenchymal views are collected in the anterior and anterolateral regions of the chest. They facilitate the assessment of the lung parenchyma (i.e., functional lung tissue). To obtain an intelligible parenchymal view, the operator must visualize the *pleural line* (see Figure 2.3a), which is a hyperechoic horizontal structure that represents the apposition of the *visceral pleura* and the *parietal pleura*. The pleural line is visible only when ultrasound waves are not obstructed by ribs. Artefacts indicating the health of lung tissue arise from the pleural line, underscoring its centrality to parenchymal lung views. Pleural views are

acquired by placing the probe in the posterolateral chest, which facilitates examination of the interface between the lung and the diaphragm. For a patient in the supine position, this view will display fluid between the pleurae of the lung if it exists. Figure 2.3 identifies orienting structures in each type of LU view. To visualize the anatomical context of lung ultrasound views and to learn more about lung ultrasound acquisition technique, we refer the interested reader to Chapter 8 in Soni *et al.* [143].



(a) A standard parenchymal view. The pleural line — which corresponds to the pleura of the lung — is visible near the top of the image. Lung tissue is beneath the pleural line. A-lines are reverberations of the pleural line. Ribs reflect most sound waves, obscuring the tissues beneath them.

(b) A standard pleural view. The liver is visible and is adjacent to the diaphragm. The lung appears dark, as it is normally aerated.

Figure 2.3: Examples of normal parenchymal and pleural lung ultrasound views, with broad localizations of orienting structures.

Ultrasound of the lung is utilized in the diagnosis of several conditions, including but not limited to pneumonia, pulmonary edema, pneumothorax, pulmonary embolism, asthma, chronic obstructive pulmonary disease, atelectasis, and fibrosis [93,143]. Numerous POCUS protocols have been described that include lung ultrasound examination. For example, the BLUE protocol is a LU examination that involves collecting videos in three zones on each side of the chest [96]. The eFAST exam is a broader examination that covers the entire torso in acute trauma settings, which includes lung views to assess for pneumothorax and hemothorax [36].

2.2.2 Lung Ultrasound Machine Learning Tasks

The studies detailed in subsequent chapters make frequent reference to LU interpretation tasks. Deep neural networks pretrained with self-supervised learning are evaluated on these tasks. Here we describe these tasks. Shorthand identifiers are provided for each task adjacent to the subsection header. Figure 2.4 provides visual examples for each binary classification task, and Figure 2.5 depicts the hierarchical relationship between them.

The datasets for each of the tasks studied in this thesis are sourced from a private ultrasound database, with the exception of the public datasets comprising a COVID-19 classification task. Details regarding the datasets are provided in the subsequent chapters.

View Classification (`View`)

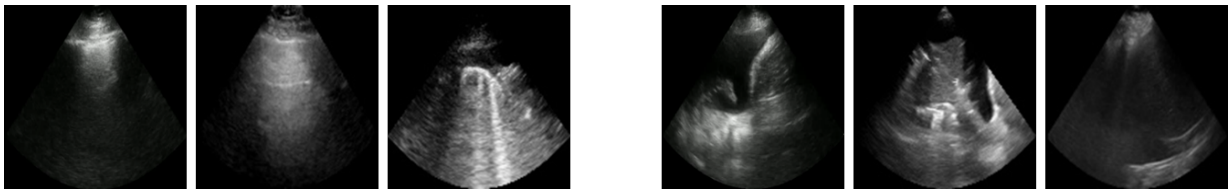
As outlined above, there are two essential types of LU views: parenchymal and pleural. Despite the existence of multiple valid sites at which to acquire each type of view on the chest, they each have a characteristic appearance. Parenchymal views exhibit the pleural line near the top of the image, along with dark areas below ribs (i.e., "rib shadows"). Pulmonary structures beneath the pleural line are typically not visualized because lungs are largely filled with air. Ultrasound waves scatter upon entering aerated mediums. Despite not being able to see the lung directly, parenchymal views are used to assess the lung tissue because the presence of certain artefacts provides clues into the health of the underlying lung tissue. Figure 2.4a provides examples of parenchymal and pleural LU images.

Consistent with prior work [158], the `View` task is a binary classification task for the purposes of this thesis. Despite this, we acknowledge that erroneous positioning of the probe could yield features of both views. We designate pleural views as the positive class.

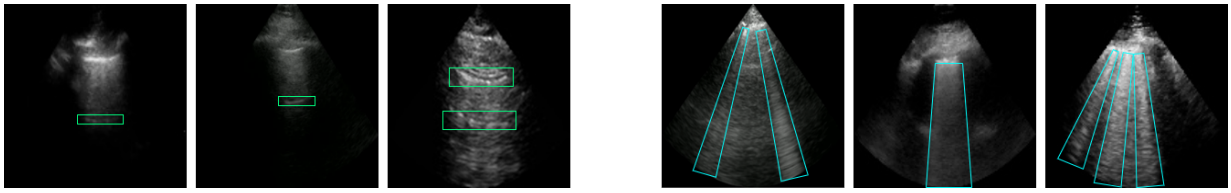
In reality, sonographers do not directly perform this classification task because they intentionally place the probe on regions of the chest that yield each type of view. However, the anatomical view is not archived in LU videos by default. Accordingly, unlabelled LU videos are generally not accompanied by view designations. As such, a fully automated LU interpretation system that is presented with LU videos without knowledge of which lung zones they were acquired from would need to distinguish between pleural and parenchymal views to determine the types of structures, artefacts, and conditions to look for.

A-lines versus B-lines Classification (`AB`)

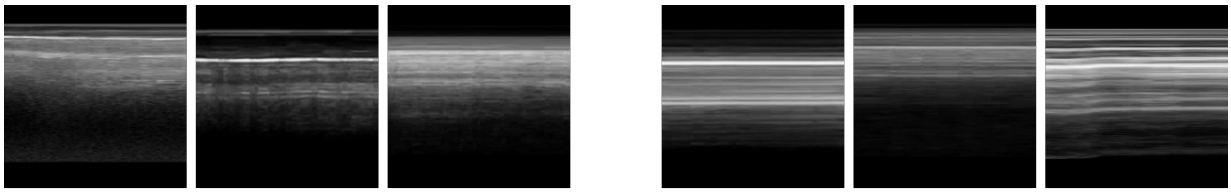
A-lines and *B-lines* are imaging artefacts that are visible on parenchymal views of the lung. Rather than corresponding to real anatomical structures, they are produced by the



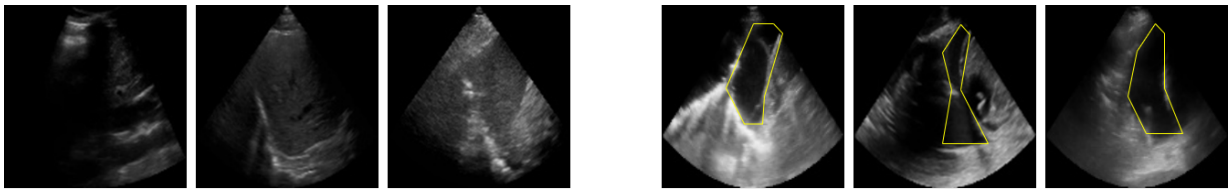
(a) B-mode ultrasound images from each class in the `View` task. **Left:** Parenchymal LU images. **Right:** Pleural LU images.



(b) B-mode ultrasound images from each class in the `AB` task. All images are parenchymal lung views. **Left:** Images containing A-line artefacts (enclosed in green boxes). **Right:** Images containing B-line artefacts (enclosed in blue trapezoids).



(c) M-mode ultrasound images from each class in the `LS` task. All images are parenchymal lung views. **Left:** Images exhibiting the seashore sign, which indicates the presence of lung sliding. **Right:** Images exhibiting the barcode sign, which indicates the absence of lung sliding.



(d) B-mode ultrasound images from each class in the `PE` task. All images are pleural lung views. **Left:** Images containing no pleural effusion. **Right:** Images containing a pleural effusion (outlined roughly in yellow).

Figure 2.4: A selection of examples from each of the lung ultrasound binary classification tasks in this thesis. For each task, images from the negative class are on the left, while positive class examples are on the right.

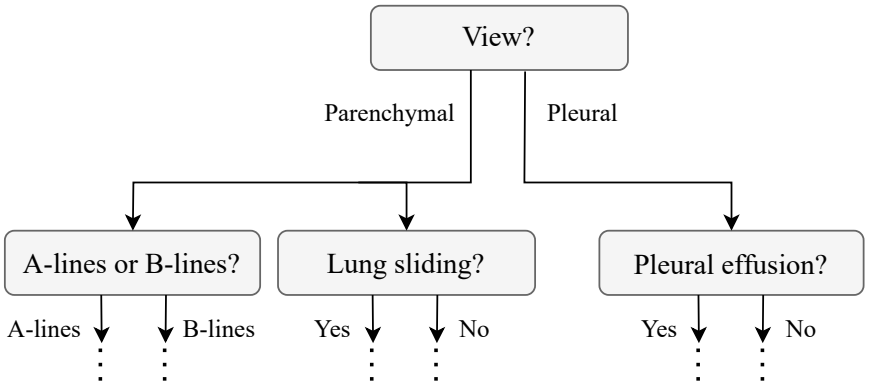


Figure 2.5: A graphical aid displaying the hierarchical relationship between core lung ultrasound classification tasks. Identification of each finding within the appropriate view can refine a differential diagnosis for patients in respiratory distress.

physical interactions of ultrasound waves in the lung.

A-lines are horizontal lines that can be seen at somewhat regular intervals below the pleural line [143]. They indicate that the tissue deep to (i.e., below) the *pleura* of the lung are aerated, which corresponds to the normal state of the lung. In fact, A-lines are reverberations of the pleural line [93]. Sound waves reflected off the hyperechoic pleural line repeatedly reverberate off the probe; the time elapsed increased with each echo, leading to successively lower A-lines on the image.

B-lines are "comet-tail" artefacts oriented along scan lines that originate from the pleural line and extend to the bottom of the ultrasound field of view [93, 143]. They obliterate A-lines, which results in the two artefacts being nearly mutually exclusive. B-lines are the result of fluid-filled interlobular septa collocated with air-filled alveoli [96]. Ultrasonic waves reverberate continuously between these tightly packed fluid-filled and air-filled regions, leading to continuous reflections that are transduced by the probe. They extend to the bottom of the field of view due to continuous emission of reflections from these regions. B-lines are a nonspecific indicator of diseased lung tissue, but they can appear in healthy individuals [143]. They may be present in a wide range of conditions, such as pneumonia, pulmonary edema, interstitial lung disease, pulmonary contusion, atelectasis, or cancer [143]. B-line appearance can be categorized by severity according to morphological characteristics [96]. Note that in rare cases, images may contain neither A-lines nor B-lines. In keeping with prior work [6], we train and evaluate models using examples that either contain A-lines or B-lines.

As in prior work [6], the AB task is considered a binary classification problem in this thesis. However, we acknowledge that there are rare cases in which A-lines and B-lines can appear on the same image. By convention, the presence of B-lines corresponds to the positive class. Figure 2.4b displays examples of each class.

Lung Sliding Classification (LS)

Recall that the pleural line is visualized on parenchymal views of the lung. On B-mode LU videos, the normal motion of the visceral and parietal pleurae when they are in contact produces a dynamic shimmering appearance of the pleural line, known as *lung sliding* [95, 143]. The absence of the lung sliding sign is suggestive of an underlying pathology, such as pneumothorax. A pneumothorax is a pathologic collection of air between the visceral and parietal pleurae of the lung. They can occur spontaneously or as a result of injury. The pleural line appears stagnant and does not shimmer when lung sliding is absent. The presence of lung sliding is useful in the acutely dyspneic patient because it rules out the possibility of a pneumothorax at the site of the probe [95].

The presence or absence of lung sliding can also be appreciated on M-mode ultrasound. M-mode images along scan lines that intersect the pleural line contain sufficient information to distinguish between these two states. Characterized by the appearance of grainy textures below the pleural line, the seashore sign indicates that lung sliding is present [92]. Conversely, the barcode sign manifests as a horizontally striated pattern below the pleural line, indicating absent lung sliding [97].

Consistent with prior work [158], we define lung sliding as a binary classification problem on M-mode LU images. Despite this, we acknowledge that there are cases in which the presence of the lung sliding sign may be indeterminate [47]. We adopt the convention of designating the absence of lung sliding as the positive class, as it is the pathological state [158]. Figure 2.4c displays examples of M-mode images depicting the presence and absence of lung sliding.

Pleural Effusion Classification (PE)

Pleural effusions are characterized as a collection of fluid between the visceral and parietal pleurae of the lung. In normal cases, the visceral and parietal pleurae should not be separated. Sonographers check for pleural effusions on pleural views of the lung. They can appear as black regions adjacent to the diaphragm, as ultrasound waves propagate through fluid [96, 143]. While simple pleural effusions appear uniformly anechoic, complex

effusions complex pleural effusions may contain patterns or dark textures [143]. In fact, echoic pleural effusions are more commonly dangerous than anechoic ones [94]. Pleural effusions may be present in a variety of conditions, such as hemothorax, empyema, heart failure, pneumonia, atelectasis, and cancer [143].

Pleural effusion detection is a binary classification task. In our investigations, we designate the presence of pleural effusion as the positive class. Figure 2.4d displays examples of images from each class.

COVID-19 Pneumonia Classification (COVID)

Lung ultrasound was used extensively throughout the COVID-19 pandemic, as researchers described ultrasonic features of pneumonia caused by the SARS-CoV-2 virus. Features observed on LU in COVID-19 pneumonia cases include bilateral B-lines, irregular pleural line, consolidations, and air bronchograms [16]. It is not trivial to distinguish COVID-19 as the etiology of a patient's pneumonia on LU compared to bacterial causes or other viral causes. While LU cannot provide a definitive diagnosis, multiple studies have reported that machine learning approaches have displayed success in distinguishing COVID-19 pneumonia from conditions that look similar on LU [5, 19, 33, 181].

In this thesis, we use two public datasets that embody slightly different multiclass classification problems. The dataset published by Born *et al.* [19] provides examples of LU examinations from three classes: COVID-19 pneumonia, non-COVID-19 pneumonia, and normal lung. conversely, the COVIDx-US dataset published by Ebadi *et al.* [43] includes examples of LU videos belonging to one of the following four classes: normal lung, COVID-19 pneumonia, non-COVID-19 pneumonia, and a broad class encompassing other lung pathologies.

Pleural Line Object Detection (PL)

As described in Section 2.2.1, the pleural line is a hyperechoic structure that manifests as a bright horizontal line near the top of parenchymal views of the lung. To confirm proper positioning of the probe, operators should visualize the pleural line centred between two ribs [143]. With correct probe placement, the operator can look for signs specific to parenchymal views, such as A-lines, B-lines, or lung sliding. Depending on image acquisition characteristics and positioning factors, multiple pleural line instances may be visible. In this thesis, the localization of the pleural line is cast as an object detection task, where the goal is to predict the presence, size, and location of bounding boxes that

enclose instances of the pleural line. Figure 2.3a provides a visual example of instances of the pleural line localized within bounding boxes.

2.3 Self-Supervised Learning

Self-supervised learning (SSL) comprises the core class of methods examined in this thesis. In this section, we provide an introduction into the basic principles and categories of SSL.

2.3.1 Motivation for Self-Supervised Learning

Significant advancements in deep computer vision have resulted in a surge of interest in applications to medical imaging. Indeed, an enormous number of publications have demonstrated the capabilities of deep learning methods in approximating diagnostic functions in radiological, histopathological, dermatological, and endoscopic imaging. Deep learning has been extensively applied in recent years to diagnostic pattern recognition tasks such as classification, object detection, and segmentation in several modalities of medical imaging [10, 39, 83, 137, 149].

Of course, methodological advances alone are insufficient to achieve nontrivial results for deep computer vision tasks. Large, labelled datasets are the major precondition for success in supervised learning problems. Fortunately, these exist some notable examples of large, open datasets for medical images that contain expert classification labels for limited sets of conditions (e.g., `CheXpert` [76]). Regrettably, large medical imaging datasets containing task- or pathology-specific labels for all constituent examples do not exist in abundance [89]. Moreover, medical imaging datasets tend to contain far fewer examples than the photographic image datasets driving much of the recent progress in computer vision [159]. Obstacles such as patient privacy concerns, private corporate interests, and the need for expert labelling, hamper the production and dissemination of such datasets [107, 166]. Occasionally, situations arise in which unlabelled datasets of medical images are available. Labelling a complete dataset requires established expertise, the cost of which dwarfs the cost of crowdsourcing labels. Furthermore, tasks such as semantic and instance segmentation require greater attention to detail, greatly increasing the labelling time per example.

SSL has emerged as a broad strategy to learn a machine learning model that produces feature representations from unlabelled data. It is particularly beneficial when only a subset of examples in a dataset have associated labels. In brief, a machine learning model

(typically a deep neural network) is trained to optimize a supervised learning objective in which the targets can be derived from the inputs themselves. In other words, a model is trained to solve a *pretext task*, which is a problem that is solvable using only the inputs and that requires no labels. *Self-supervised pretraining* refers to the optimization of a self-supervised objective to obtain a model capable of producing meaningful feature representations that capture salient information available in the inputs. The learned parameters of the pretrained model may then be used to initialize a new model that can be trained to solve a more specific supervised learning problem for which labelled data is available. Figure 2.6 portrays an example of the steps undertaken to pretrain a model using SSL to learn representations of animal photographs, prior to training a multiclass animal classifier.

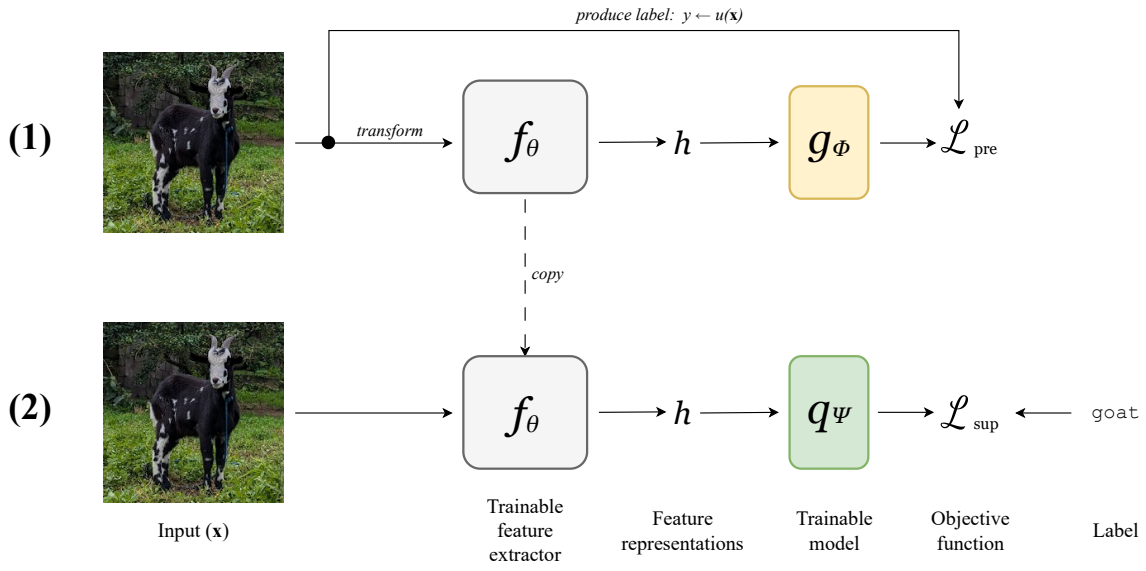


Figure 2.6: Example of a typical SSL workflow, with an application to animal classification. **(1)** *Self-supervised pretraining:* A parameterized model $g_\phi(f_\theta(\mathbf{x}))$ is trained to solve a pretext task using only unlabelled photographs of animals. The labels for the pretext task, y, are determined from the inputs themselves according to some function $u(\mathbf{x})$, and the model is trained to minimize the pretext objective $\mathcal{L}_{\text{pre}}$. At the end of this step, f_θ should output useful feature representations. **(2)** *Supervised fine-tuning:* Parameterized model $q_\psi(f_\theta(\mathbf{x}))$ is trained to solve the supervised learning task of animal classification using animal labels. Note that the previously learned f_θ is reused for this task, as it produces feature representations for the animal dataset.

SSL is well-suited to facilitate the advancement of automated diagnostic tasks with

radiological images, as vast quantities of historical data are available in picture archiving and communication systems at healthcare institutions worldwide, but labels may not be present. Although accompanying radiological reports may exist in electronic medical records, it is laborious to devise classification labels from unstructured text. Furthermore, reports may not explicitly identify all relevant negative findings for conditions of interest, opting to omit descriptions of normality. Matters are especially complicated in the context of segmentation tasks. Regardless, it is rare to encounter a fully labelled retrospectively acquired dataset. It is often necessary for experts to label at least a fraction of the dataset that has not been built with human annotation labour. Expert labelling may be prohibitively expensive in terms of monetary cost and/or time. SSL pretraining can therefore materially reduce the burden on experts to label entire radiological datasets.

Although SSL is a widely recognized strategy for productively using unlabelled data, there are other categories of machine learning that combine unsupervised learning with unlabelled data with supervised learning. *Semi-supervised learning* is a category of methods that involves training machine learning algorithms to predict labels according to a task of interest that uses labelled and unlabelled examples. Classically, the objective to minimize contains separate supervised terms for labelled data, unsupervised terms for unlabelled data, and regularization terms [173]. There are many forms of semi-supervised learning that solve the aforementioned objective in various ways. For example, *Self-training* is a category of semi-supervised learning in which the predictions of a neural network previously trained on labelled examples are used to produce pseudo-labels for unlabelled examples, which are then used to train a new network using supervised learning [3]. The process may then be repeated. Semi- and self-supervised learning are both applicable in settings where unlabelled examples outnumber labelled examples. A core difference between the two is that self-supervised learning is conducted in two successive phases: (1) unsupervised pretraining and (2) supervised learning; on the other hand, semi-supervised learning constitutes task-specific training. Self-supervised pretraining is typically task-agnostic, while semi-supervised learning is task-specific. DNNs pretrained with self-supervised learning extract generic representations. This thesis focuses on self-supervised learning because task-agnostic DNNs can be re-used for multiple tasks, thereby entailing the design of efficient automated LU multi-task interpretation. Chapter 3 further elucidates the need for task-agnostic DNNs for LU.

2.3.2 Self-Supervised Learning Preliminaries

In unsupervised *representation learning*, machine learning models are trained to produce d-dimensional representations of inputs that are useful for some task(s), where d is much

smaller than the dimensionality of the inputs. SSL is a form of representation learning in which the objective function is formed from a pretext task whose solutions are easily obtainable from unlabelled examples. SSL distinguishes itself from supervised learning in that the objective does not depend on labels for any specific task. SSL aims to derive low-dimensional representations for examples; however, it is distinct within unsupervised learning in that it involves optimizing supervised learning objectives.

The goal of SSL is to learn a *feature extractor* (also known as a *backbone* or *encoder*) that can extract high-level representations from examples. The weights of the feature extractor may then be applied to subsequent supervised learning tasks for which labels are available (often referred to as *downstream tasks*). The weights of the feature extractor may be kept stagnant or fine-tuned in the downstream learning problem. To gain intuition into the advantage of learning representations with SSL, consider the following example. Suppose a toddler is seeing various kinds of fruits for the first time. Without any feedback or external knowledge, she discovers attributes of the fruits that distinguish them from others, such as colour and shape. Later, when she is taught to identify fruits by name in preschool, she applies her previously acquired knowledge about the distinctive features of fruits to help her classify them (e.g., limes are green and round). It is likely that they have an advantage over classmates who did not eat fruit at home.

More concretely, suppose that we have a dataset of examples, $\mathcal{X}$. A *pretext task* is formulated that is solvable with knowledge of the examples. Note that the task may be defined for one or more examples. Solutions for the pretext task are taken as the labels for a self-supervised problem. An objective is defined that appropriately measures the performance of a learner at solving the pretext task.

In the context of computer vision, a backbone model, $f_\theta : \mathbb{R}^{h \times w \times c} \to \mathbb{R}^d$, is defined that maps $h \times w \times c$ images to a d-dimensional representation. f_θ is typically a deep neural network, parameterized by θ, whose architecture embodies an inductive bias amenable to the equivariance and invariance relationships inherent to the dataset, such as a CNN. The objective is computed from the output of a secondary function $g_\phi : \mathbb{R}^d \to \mathbb{R}^e$, where g_ϕ is a neural network with parameter ϕ. The parameters are optimized to reduce the value of the pretext objective $\mathcal{L}_{\text{pre}}$.

$$(\theta^*, \phi^*) = \arg\min_{\theta,\phi} \mathcal{L}_{\text{pre}}(g_\phi(f_\theta(\mathbf{x}))) \tag{2.1}$$

For the animal classification example in Figure 2.6, suppose that images of animals are passed to a CNN feature extractor f_θ. The resulting feature representations $\mathbf{h}$ are passed to multilayer perceptron (MLP) g_ϕ, the output of which is used to compute $\mathcal{L}_{\text{pre}}$, which

quantifies performance on the pretext task.

After the objective is optimized, g_ϕ is customarily discarded. The backbone f_θ may then be applied for a subsequent supervised learning problem, as $f_\theta(\mathbf{x})$ is a nontrivial representation of $\mathbf{x}$. For a supervised learning task with examples $\mathcal{X}'$ (originating from an identical or similar distribution as $\mathcal{X}$) and corresponding labels $\mathcal{Y}$, a new model head $q_\psi : \mathbb{R}^d \to \mathbb{R}^{\dim(y)}$ is initialized. q_ψ receives feature representations $\mathbf{h}$ as input. The model $q_\psi(f_\theta(\mathbf{x}))$ is trained to minimize a loss function with respect to the labels. At this stage, θ may be held constant or fine-tuned via transfer learning. Note that it is possible that the pretrained weights θ may constitute a useful initialization for multiple downstream supervised learning problems that share similar features.

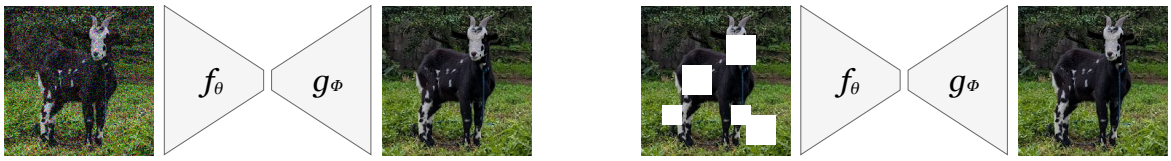
2.3.3 Taxonomy of Self-Supervised Learning

The major difference between various self-supervised pretraining methods is the choice of pretext task and its optimization. Here we enumerate some broad categories of SSL methods. The intention is to provide the reader with a high-level understanding of the main approaches to SSL that may be useful when describing specific studies in the subsequent chapters. These approaches are often trialled on natural images first, likely due to the high availability of benchmark datasets and broad applicability.

Generative Methods

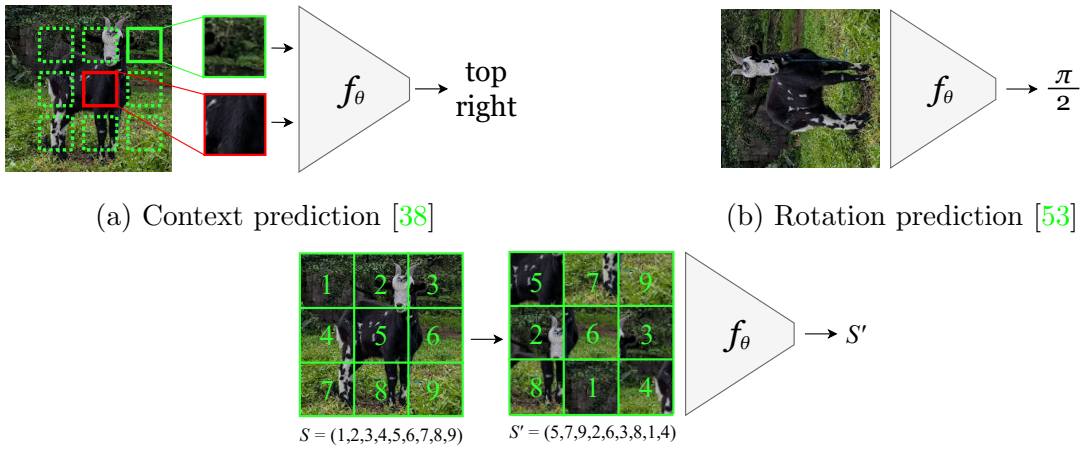
Several SSL pretext tasks are built around generating samples. The output of g_ϕ is an entire image or a fragment of an image. Note that generative SSL methods are not to be confused with general generative methods in machine learning (e.g. generative adversarial networks [55], denoising diffusion models [69]), where the focus is on image generation and not necessarily on producing a feature extractor. Generative SSL methods often employ an *encoder* network that learns rich feature representations. The feature representations are sent to a secondary network, frequently referred to as a *decoder*. In a self-supervised context, g_ϕ is the decoder and f_θ is the encoder, where the latter is retained for downstream supervised learning. Many generative tasks are reconstructive, in that the goal is to recover a corrupted version of an image. An early example of a reconstructive approach to SSL is the denoising autoencoder [161] (Figure 2.7a). In the image colourization task, coloured images are generated from grayscale images, which is made possible by the availability of a dataset of coloured images [186]. Inpainting redacted patches of images is another example of a reconstructive pretext task [124] (Figure 2.7b).

The masked image modelling (MIM) approach to SSL comprises a family of generative methods where models are trained to predict redacted portions of images, typically denominated in square patches. These approaches are resonant of the masked language modelling techniques popularized in natural language processing. Vision transformers are the architecture primarily employed because images are formulated as a sequence of tokens corresponding to patches. The Masked Autoencoder (MAE) [66] and SimMIM [168] methods are two common approaches to MIM that employ an encoder-decoder architecture to predict masked patches in images. In MAE, the encoder produces representations for unmasked patches, and the decoder is trained to reconstruct the image. The network is trained using the mean squared error loss computed on the masked patches. SimMIM is similar to MAE, however it differs in that it feeds mask tokens to the encoder, it uses a linear decoder and is typically trained with mean absolute error [168].



(a) Denoising [161] (b) Inpainting [124]

Figure 2.7: Examples of generative SSL pretext tasks



(a) Context prediction [38] (b) Rotation prediction [53]

(c) Patch permutation prediction (jigsaw) [119]

Figure 2.8: Examples of predictive SSL pretext tasks

Predictive Methods

Many custom pretext tasks have been proposed for computer vision that involve learning a specific transformation applied to images. A stochastic transformation is applied to each example, and the learner's task is to predict or to undo the transformation. For instance, the context prediction task is defined as the problem of predicting the relative location of random image patches from unlabelled images (Figure 2.8a) [38]. In the rotation prediction task, a random rotation is applied to an image and the learner must infer which rotation was applied (Figure 2.8b) [53]. The jigsaw task is the unscrambling of a random permutation of all the rectangular patches in an image (Figure 2.8c) [119]. Generally, the label is defined as the transformation that was applied to the image. The transformation may be stochastic in that its parameters may be sampled from some underlying distribution (e.g., the angle of a rotation being sampled from a categorical distribution over predefined angles).

A criticism of transformation prediction methods is that they may not be generally applicable to downstream tasks because the pretext tasks are formulated using specialized heuristics [29]. *Joint embedding self-supervised learning* is a category of predictive methods where the pretext tasks are centred around learning to extract similar feature vectors for semantically related images. Framed succinctly, the goal is to learn representations that are invariant to non-meaningful differences. It has evolved as an approach for learning feature representations with fewer assumptions regarding the usefulness of particular tasks. In this paradigm, $g_\phi(\mathbf{h})$ is a neural network that outputs vector *embeddings*, which need not have the same dimension as the representations $\mathbf{h}$[1]. In joint embedding SSL methods, the neural network g_ϕ is commonly referred to as a *projector*. It is common for pairs of semantically related images to be produced by producing two distorted views of each image in a batch. Figure 2.9 provides an overview of the core components of joint embedding SSL. There are two major categories of joint embedding methods: *contrastive learning* and *non-contrastive learning*.

The goal of contrastive learning is to produce embeddings $\mathbf{z}_i$ that are very close (as measured by some distance function for *positive pairs* of examples and very far for *negative pairs*. In general, a positive pair is any pair of images that satisfies a *pairwise relationship* that indicates semantic similarity between the images. Negative pairs of images do not satisfy the pairwise relationship and are typically dissimilar. SimCLR [29] is an example of a contrastive learning SSL method in which positive pairs are distorted versions of the same image and negative pairs are distorted versions of distinct images. The weights θ

[1]To further clarify nomenclature, this thesis refers to the vector output of f_θ as *representations*, while *embeddings* are defined as the vector output of g_ϕ. The definitions of these terms are consistent with prior literature [12, 29, 180].

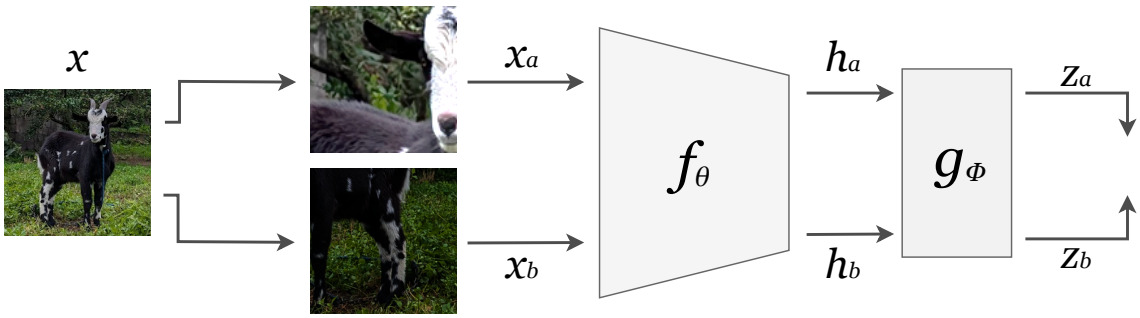


Figure 2.9: A depiction of the forward pass for a positive pair in a standard joint embedding method. Each image in a batch is subjected to stochastic data augmentation twice, producing distorted views $\mathbf{x}_a$ and $\mathbf{x}_b$, which are passed through the feature extractor f_θ to yield feature representations $\mathbf{h}_a$ and $\mathbf{h}_b$. The projector g_ϕ transforms $\mathbf{h}_a$ and $\mathbf{h}_b$ into embeddings $\mathbf{z}_a$ and $\mathbf{z}_b$, respectively. Typically, the objective $\mathcal{L}$ is optimized to maximize the similarity of $\mathbf{z}_a$ and $\mathbf{z}_b$.

and ϕ are optimized such that the embeddings are close and far for positive and negative pairs, respectively. SimCLR uses the NT-Xent loss function, which encourages similarity between embeddings of positive pairs while pushing negative pairs apart in embedding space. Consider a set of $2N$ samples constructed by producing two distorted versions for a batch of N examples, where embeddings $\mathbf{z}_i$ and $\mathbf{z}_j$ correspond to the embeddings of a positive pair. The NT-Xent loss computed for this positive pair is given as

$$\mathcal{L}_{i,j} = -\log \frac{e^{\text{sim}(\mathbf{z}_i,\mathbf{z}_j)/\tau}}{\sum_{k=1}^{2N-1}(1-\delta_{ki})e^{\text{sim}(\mathbf{z}_i,\mathbf{z}_k)/\tau}} \tag{2.2}$$

where $\text{sim}(\cdot,\cdot)$ is a similarity function (typically cosine similarity), δ_{ki} is the Kronecker delta evaluated at values k and i, and τ is a temperature parameter [29]. The SimCLR loss function is the sum of $\mathcal{L}_{i,j}$ for each positive pair in the batch. To produce distorted versions of images, a series of data augmentation transformations is applied, where the parameters of the transformation are sampled from a probability distribution. Common examples of transformations include cropping, affine transformations, noise addition, and adjustments to brightness, contrast, and hue. It is thought that the projector g_ϕ is responsible for removing noise introduced by positive pair misalignment (which can occur in the context of data augmentation), which may explain why representations outputted by the backbone are more useful during fine-tuning than the embeddings outputted by the projector [17]. Empirically, the inclusion of a projector and its removal following pretraining were observed

to be critical to SimCLR's performance [29]. Other notable examples of contrastive learning in SSL include MoCo [67], and PIRL [113]. Note that, although data augmentation is the customary approach to constructing positive pairs, the pairwise relationship can be formulated by exploiting attributes linking semantically related instances, when possible.

Originally, it was thought that a weakness of contrastive learning was that its reliance on negative pairs necessitated large batch sizes [59, 180]. Non-contrastive learning emerged as an alternative approach relying only on positive pairs. Different transformations are applied to the same image to produce multiple views. f_θ and g_ϕ are optimized to produce embeddings that are robust to the possible views entailed by the transformation distribution, through minimizing the distance between the embeddings of positive pairs. Various strategies have been devised to avoid the problem of *information collapse*, where models learn the trivial solution of indiscriminately outputting zero vectors as embeddings. Barlow Twins is an example of non-contrastive learning [180]. Suppose that $\mathbf{Z}_1, \mathbf{Z}_2 \in \mathbb{R}^{N \times d}$ are batches of d-dimensional embeddings where $\mathbf{Z}_{1i}$ and $\mathbf{Z}_{2i}$ are the embeddings of a positive pair. Then the Barlow Twins loss function is defined as

$$\mathcal{L}_{\text{BT}} = \sum_i (1 - \mathcal{C}_{i,i})^2 + \lambda \sum_i \sum_{j \neq i} \mathcal{C}_{i,j}^2 \tag{2.3}$$

where $\mathcal{C}$ is the cross-correlation matrix of paired embedding batches $\mathbf{Z}_1$ and $\mathbf{Z}_2$ normalized across the batch and λ is a scaling factor. The first term encourages the minimization of the distance between the embeddings of positive pairs. The second term preserves variance in the embeddings across the batch. Inspired by Barlow Twins, the third term promotes the decorrelation of embeddings across dimensions. For paired embedding batches $\mathbf{Z}_1$ and $\mathbf{Z}_2$, its loss function is given as

$$\mathcal{L}_{\text{VICReg}} = \lambda s(\mathbf{Z}_1, \mathbf{Z}_2) + \mu\big(v(\mathbf{Z}_1) + v(\mathbf{Z}_1)\big) + \nu\big(c(\mathbf{Z}_1) + c(\mathbf{Z}_2)\big) \tag{2.4}$$

where $s(\cdot, \cdot)$ is mean squared error; $v(\cdot)$ is a hinge function that promotes minimum batch-wise variance for each embedding dimension; $c(\cdot)$ is a sum of the off-diagonal elements of the covariance matrix of the argument; and λ, μ, and ν are coefficients that control the influence of each term. Other non-contrastive methods have been proposed that integrate distinct backbones during pretraining. For example, the Bootstrap Your Own Latent (BYOL) method uses an online and target network, where the online network predicts the output of the target network whose weights are an exponential moving average of the online network's weights [59]. The ensuing asymmetry discourages information collapse. SimSiam [31] and DINO [24] are two additional examples of non-contrastive methods that

involve two distinct backbones where one predicts the output of the other.

2.3.4 Theoretical Support

Until recently, SSL publications commonly introduced novel methods guided by intuition. Some researchers have since attempted to explore the properties of SSL pretraining to better understand why they deliver such benefits and to ascertain conditions under which they will succeed.

Efforts in attempting to understanding the efficacy of optimizing performance on pretext tasks in learning downstream tasks are growing. Lee *et al.* [88] provided guarantees for the improved sample efficiency of pretraining with a reconstructive pretext task, in scenarios where the inputs and pretext target are conditionally independent of the labels and a latent variable. Dropping the conditional independence assumption, HaoChen *et al.* [65] defined a contrastive loss based on spectral decomposition and derived performance guarantees for linear classifiers trained on the feature representations from the pretraining phase. Balestriero and LeCun [11] developed an amalgamated lens for viewing contemporary contrastive and non-contrastive approaches as spectral analysis. They demonstrated that a selection of SSL methods (namely Barlow Twins [180], VICReg [12], and SimCLR [29]) are optimal choices for solving downstream tasks as long as the relation between labels is included in the relationship between positive pairs [11]. Practitioners in all domains of computer vision are therefore encouraged to ensure that their choice of pretext task aligns appropriately with the label distribution. Those applying SSL pretraining for radiological imaging tasks should consider these results when selecting a pretext task.

2.4 Self-Supervised Learning in Ultrasound

2.4.1 Literature Search Methodology

A literature survey was conducted to consolidate and evaluate studies quantifying the benefit of self-supervised pretraining in the automation of diagnostic tasks concerning radiological images. A set of potentially qualifying publications as of November 2022[2] was found by searching the following four databases: Scopus, IEEE, ACM, and PubMed. Queries were designed to cast a wide net, including all studies whose titles, abstracts, keywords, or

[2]The literature survey was written in 2023. The present background chapter has been expanded to address recent publications relevant to SSL in ultrasound.

bodies mention medical images, CT, MRI, X-ray, ultrasound *and* self-supervised learning or contrastive learning. The search returned a total of 1226 results, which was reduced to 778 unique studies by removing duplicate and completely irrelevant papers.

Exclusion criteria were applied to the results to narrow down the body of literature to those assessing the impact of self-supervised pretraining. Studies were excluded if any of the following conditions were met:

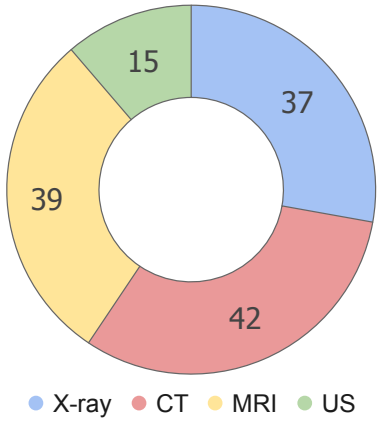
- The study was not concerned with the radiological modalities within the scope of the survey (i.e., MRI, CT, X-ray, ultrasound).
- SSL objectives were presented in the context of semi-supervised learning, which is a family of machine learning techniques related to SSL. Like SSL, semi-supervised methods exploit unlabelled data. However, it is concerned with the simultaneous optimization of the supervised learning task of interest and an unsupervised objective.
- The study was a preprint or had not been published in a peer-reviewed periodical at the time of the literature search.
- The study applied self-supervised learning for a diagnostic task but did not compare performance on their downstream supervised learning task to a baseline (e.g., weights trained from scratch or initialized using weights pretrained on `ImageNet` [35]).

The result was a collection of 124 studies. Figure 2.10a visualizes the distribution of these papers by imaging modality. As shown in the figure, there are considerably less self-supervised pretraining publications geared toward ultrasound tasks than for X-ray, CT, or MRI. Figure 2.10b compares the number of papers in the survey published per year, reflecting the increasing interest and progress in SSL during recent years.

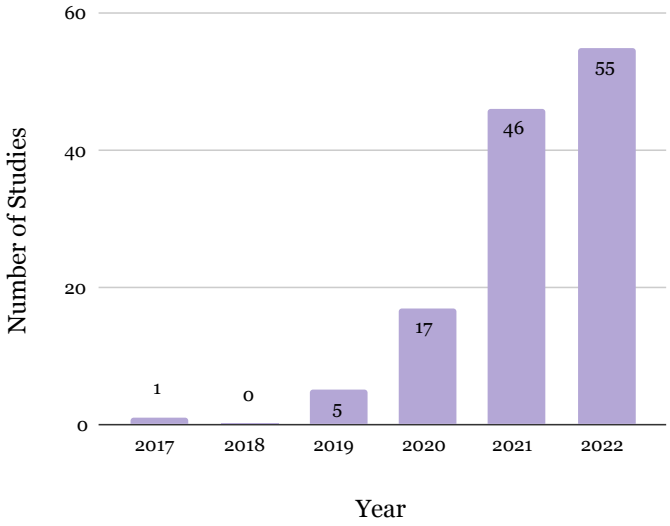
The survey directly addressed all included studies. We focused on studies that concentrate on common downstream tasks. Additionally, we attempted to highlight studies featuring replicable methods, as indicated by evaluation on public datasets and open-source availability of experiment code. The survey content included in this thesis is limited to studies concerning ultrasound imaging. We refer the interested reader to the published survey [153] to access results for the other radiological modalities.

2.4.2 Applications in Ultrasound Imaging

Here we describe evidence from the literature regarding the pretraining machine learning models for diagnostic tasks with ultrasound examinations. As outlined in Section 2.4.1,



(a) The number of studies presenting experiments for each of the four modalities covered in this review: X-ray, CT, MRI, ultrasound. Note that some studies investigated multiple of the above modalities.



(b) The number of publications by year investigated in the survey, reflecting a growing interest in SSL for radiological imaging.

Figure 2.10: Breakdown of the papers included in the survey by (a) imaging modality and (b) year of publication.

considerably fewer publications have explored self-supervised pretraining for ultrasound than for the preceding three types of radiological imaging. This section has been augmented to include studies that have surfaced since the publication of the literature survey.

Breast Ultrasound Lesion Characterization

Nguyen *et al.* [118] explored the efficacy of BYOL [59] for the classification of breast ultrasound images from the public `BreastMNIST` dataset [172] as either normal, containing benign tumours, or containing malignant tumours. Although the paper contains terminological errors, it provides a benchmark for a non-contrastive method on a public dataset. They found that pretraining with BYOL resulted in worse test performance than randomly initialized or `ImageNet`-pretrained weights. Perek *et al.* [126] arrived at a similar conclusion when trialling MoCo with a private dataset [67]. Proposing a video-specific pretext task instead, Lin *et al.* [98] pretrained an encoder-decoder architecture to restore an ul-

trasound video after randomly masking out entire frames and patches in the remaining frames. Upon performing semi-supervised fine-tuning for benign versus malignant lesion classification on a private dataset, masked video pretraining yielded 1% greater accuracy compared to random initialization. Focusing instead on breast lesion semantic segmentation, Mishra *et al.* [113] pretrained an encoder-decoder to perform a deterministic edge detection or segmentation task that does not require machine learning. They performed experiments using two publicly available datasets (`BUSI` [1] and `UDIAT` [175]) and observed that SSL improved performance, with the gap increasing with less labelled training data availability. However, it is unclear which pretext task they selected for their downstream experiments. Chen *et al.* [33] proposed Ultrasound Contrastive Learning (USCL) — a contrastive learning method that constructs positive pairs as weighted sums of images within the same ultrasound video. Three images from the same video are randomly sampled and arranged in chronological order. Positive pairs are constructed from weighted pixel-wise sums of the first two images and from the last two images. After pretraining on liver and lung ultrasound images, the authors reported improved objected detection and segmentation performance for fine-tuned models on `UDIAT` [175] compared to initialization with ImageNet-pretrained weights. A subsequent study proposed Meta-USCL, which builds on USCL by training a separate network that learns to weigh positive pairs concurrent with self-supervised pretraining [32]. The authors reported slightly improved results on `UDIAT` compared to USCL, but similar results on BUI [131], a breast ultrasound classification dataset. The HiCo contrastive learning method applied a softened version of the InfoNCE objective to intermediate CNN feature maps for positive pairs at the same network block and across multiple network blocks [182]. When evaluating on a union of the `BUSI` and `BUI` datasets, the authors reported a slight improvement in breast lesion classification. Xu *et al.* [170] adopted a MIM approach, reporting greater performance than USCL and HiCO on the aforementioned joint `BUSI-BUI` dataset. All pretraining was conducted using tongue ultrasound images from the public `UXTD` dataset [44]. Rahman and Patel [130] proposed UltraMAE, which builds on the MAE method [66] by comparing reconstructed regions to ground truth images with reduced noise. The authors reported improved performance on `BUSI` compared to full supervision, USCL [33], and another contrastive method [13].

Echocardiography Tasks

SSL has been cited as useful for a variety of echocardiography interpretation tasks. Anand *et al.* [4] sought to establish the performance of ubiquitous contemporary joint embedding SSL methods for the task of view classification (e.g., SimCLR [29], MoCoV2 [30],

BYOL [59], DINO [24]). Not only did they find that pretraining outperformed random and `ImageNet`-pretrained initialization, but they demonstrated that pretraining with more unlabelled data widened the performance gap. SimCLR and BYOL pretraining have been investigated for the task of left ventricle segmentation. Saeed *et al.* [132] observed that SimCLR pretraining generally resulted in the best Dice score, but the difference was small across label availability fractions. Surprisingly, BYOL pretraining generally resulted in worse performance than full supervision. The results appeared to be consistent across two public datasets: `EchoNet-Dynamic` dataset [121] and `CAMUS` [87]. To reduce redundancy of pretraining examples, they chose to use one randomly selected frame per clip during pretraining, despite using two labelled frames per clip for the downstream task (one each for end-systole and end-diastole); it is possible that using more frames during pretraining may have improved performance. Dezaki *et al.* [37] devised a multifaceted pretext task customized for echocardiograms that consists of (1) reordering shuffled triplets of contiguous frames, (2) minimizing embeddings for continuous frames and maximizing embeddings for temporally distance frames, and (3) minimizing the differences between embeddings of frames from multiple views corresponding to the same point in the cardiac cycle. Although fully supervised learning matched self-supervised pretraining when using all labels, SSL greatly improved performance when less labels were available. They observed similar results when evaluating on `EchoNet-Dynamic`.

Ultrasound Assessment of Thyroid Nodules

Ultrasound is often employed to assess thyroid nodules for possible malignancy. Hu *et al.* [72] pretrained CNN backbones on private ultrasound datasets using a combination of a context reconstruction pretext task and a discriminator model. The authors reported improved performance on the public `DDTI` [125] dataset for segmenting thyroid lesions compared to; however, the metric quantifying segmentation performance is unclear. They also found that augmenting the discriminator's output with Digital Imaging and Communications in Medicine (DICOM) tags during pretraining improved performance. Zhao and Yang [190] pretrained a classifier to distinguish between benign and malignant nodules, using the public `TN-SCUI2020` [141] dataset. They integrated prior medical knowledge into their contrastive pretext task, which sought to minimize the differences between embeddings of handcrafted radiomics features and the original ultrasound image. Their method outperformed random initialization and pretraining with generic pretext tasks. Xiang *et al.* [167] also devised a custom pretext task for this problem, characterized by thyroid ultrasound modality classification. In addition to B-mode ultrasound, their downstream model received corresponding images from three ultrasound modalities, noting su-

perior performance on their private dataset when pretraining as opposed to random or `ImageNet`-pretrained initialization. Guo *et al.* [60] focused on the related downstream task of grading nodules according to the widely adopted TI-RADS [148] system. Kang *et al.* [80] proposed an ultrasound-specific improvement on the MAE method [66] that is characterized by adding an image blur preprocessing step. Evaluating on a private dataset, they observed improved performance in distinguishing benign from malignant thyroid lesions compared to standard MAE pretraining and full supervision. Lastly, Wang *et al.* [162] performed contrastive pretraining using a private dataset that contained multiple views per patient; as a result, they used different types of views of thyroid nodules to sample positive pairs. They concluded that cross-view positive pairs improved performance on nodule classification and segmentation.

Ultrasound Obstetric Tasks

Jiao *et al.* [79] described a custom ultrasound-specific pretext task consisting of predicting the order of 4 shuffled frames and predicting the continuous parameters of random affine transformations applied to the frames, which resulted in an improvement over training from scratch for the task of fetal plane detection. Chen *et al.* [27] observed a similar result for the same task. They adopted a reconstructive task, corrupting images by swapping randomly sampled square patches of pixels. Focusing instead on segmenting the utero-placental interface, Qi *et al.* [129] pretrained a feature extractor for a customized jigsaw pretext task in which the permuted patches were sampled from regions of images that intersected the labelled region of interest. Results indicated marginal improvement with pretraining, but the custom pretext task did not outperform Jigsaw [119] for the majority of feature extractors studied. Of note is the fact that their pretext task cannot be considered SSL because, by definition, the pretext task is solved in the absence of labels. Developed for 3D CNNs and transformers, the Contrastive Rubik's Cube Recovery method [185] combined permutation and rotation prediction for subvolumes of ultrasound videos and contrastive learning on distorted subvolume features. The authors reported improved performance on fetal plane detection and anatomy classification on private datasets, compared to other SSL methods and fully supervised learning. Lastly, Benjamin *et al.* [14] assessed several SSL pretraining methods for cardiac fetal plane classification on a private dataset, finding that some SSL methods resulted in improved performance compared to full supervision starting with ImageNet-pretrained weights.

Lung Ultrasound Tasks

Most related to the work in this thesis are studies that trialled SSL methods for problems in LU. Overwhelmingly, studies evaluated their methods by assessing model performance on public COVID-19 classification datasets. The authors of USCL reported an improvement in COVID-19 classification on the public `POCOVID-Net` dataset [19], compared to ImageNet pretraining. Note that USCL was originally proposed as a semi-supervised learning method that jointly optimized cross-entropy using ground truth labels. However, the authors also showed that self-supervised pretraining using only the contrastive term also improved performance on `POCOVID-Net`. Basu *et al.* [13] proposed an ultrasound-specific contrastive pretext task that considered temporally separated frames from the same video as negative pairs, in addition to inter-video pairs. Positive pairs were frames separated temporally by no more than a predefined constant number of time steps. Further, they imposed a curriculum by gradually decreasing the minimum temporal distance constituting an intra-video negative pair. Intra-video negative pairs are important to consider because the anatomical context of an ultrasound video may differ dramatically throughout its duration. However, the authors did not address how negative pair sampling would be considered for cases where the probe is kept stationary throughout the video. They observed an improvement on the `POCOVID-Net` dataset compared to USCL and an assortment of generic joint embedding methods (e.g., SimCLR [29]). Meta-USCL [32] and HiCo [182] also evaluated their contrastive pretraining methods on `POCOVID-Net`, each reporting improved performance compared to USCL. Focusing instead on video-level self-supervision, Chen *et al.* [28] observed that pretraining a 3D CNN on unlabelled LU videos using the BYOL method [59] led to a small improvement in lung consolidation classification compared to supervision alone. However, these performance gains were substantially greater when fewer labelled videos were included in the training set.

Other Ultrasound Diagnostic Tasks

Liu *et al.* [99] pretrained an encoder-decoder model for the downstream task of classifying gastrointestinal stromal tumours from endoscopic ultrasound images, observing greater average performance than full supervision with `ImageNet`-pretrained initialization (albeit with greatly overlapping confidence intervals). Interestingly, they leveraged thyroid and breast ultrasound datasets for pretraining. Zhou *et al.* [195] found random permutation prediction to be a helpful pretext task for rheumatoid arthritis grading on ultrasound; however, the approach required manual region of interest labelling. Basu *et al.* [13] (introduced in the preceding section) evaluated their approach on a private dataset for gallbladder

malignancy detection, citing performance superior to `ImageNet`-pretrained initialization, SimCLR [29], and MoCoV2 [30]. Lastly, Liu *et al.* [101] evaluated several SSL methods on a large multi-task, ultrasound dataset that they assembled from two institutions and made publicly available. Their `US-MTD120K` dataset includes thyroid, breast, liver, and fetal ultrasound images and is labelled for 28 distinct classes. After evaluating several SSL methods, they found that some SSL methods improved performance compared to full supervision while some did not. It is worth noting that they evaluated SSL methods with linear classifiers trained atop frozen backbones. Moreover, it remains unclear how train/test splits were conducted, which is important because many images in the dataset were copies transformed offline or originated from the same patient.

2.4.3 Key Literature Survey Findings

Comparison to Random Initialization

The previous section illustrated the usefulness of self-supervised pretraining in deep learning for diagnostic tasks with ultrasound images. For each of the four major modalities investigated, there are multiple studies that report an improvement in downstream performance metrics when initializing feature extractors with SSL-pretrained weights, generally compared to random weight initialization in the fully supervised setting.

In most cases, studies demonstrated that pretraining was useful either as a first step using all labelled data, or that pretraining was particularly helpful in low label availability settings. When labels are completely available for a downstream task, there is a wide variation in the change in performance on test data. Some studies report minimal or no improvement [62, 99, 129], while others report significant gains [7, 86, 105, 122]. Naturally, there are myriad reasons for such variability, including dissimilar pretext tasks, evaluation protocol differences, modality-specific noise, dataset volume and diversity, and downstream task difficulty.

The results of this review overwhelmingly suggest that pretraining with self-supervised learning is likely to result in improved performance on downstream supervised learning tasks, compared to randomly initialized supervised learners. Practitioners should consider trialling pretrained feature extractors during model development.

The Effectiveness of `ImageNet`-pretrained Weights

The vast majority of the methods explored in this review compared their pretrained models to the fully supervised setting where weights are randomly initialized. Many also compared

the results of their custom SSL method to previously proposed SSL methods that are not geared toward any specific imaging distribution. However, a fraction of studies compared their pretrained feature extractors to `ImageNet`-pretrained weights. It is reasonable to compare against `ImageNet`-pretrained weights because several medical computer vision models are initialized with `ImageNet`-pretrained weights [9]. Indeed, many studies reported that `ImageNet`-pretrained weights fared better than random initialization, making them a stronger baseline against which to compare [9, 162]. Crucially, multiple studies reported cases where 2D CNNs or vision transformers did not appreciably outperform `ImageNet`-pretrained initialization [45, 62, 62, 118, 123, 194]. We therefore advise authors of future SSL studies to compare their approaches to fully supervised baselines with random initialization *and* `ImageNet`-pretrained initialization where applicable.

A frequently absent experimental setting is the assessment of the effect of initializing feature extractors with `ImageNet`-pretrained weights *prior* to self-supervised pretraining. The small set of studies that performed this comparison observed that the best performance in downstream radiological imaging interpretation tasks was achieved by setting the initial weights of the feature extractors to `ImageNet`-pretrained weights [9, 108, 162]. SSL studies with medical images should include this experiment in their evaluation protocol. It is, however, necessary to acknowledge that publicly available `ImageNet`-pretrained weights do not exist for all feature extractor architectures (e.g., 3D CNNs), and that fully supervised pretraining can be prohibitively expensive.

Acknowledging the utility of `ImageNet`-pretrained weights, the studies in this thesis compare all self-supervised models to fully supervised learning using publicly available DNNs pretrained on `ImageNet`.

Utility of Self-Supervised Learning in Low-Label Settings

Aside from direct comparisons to fully supervised models using the same dataset, several studies have established the benefit of self-supervised pretraining in scenarios where labels are not provided for all available examples. Typically, such claims are established by comparing performance of fully and self-supervised models at different fractions of label availability, limiting the amount of training data available for supervised fine-tuning on a downstream task [9, 27, 37, 82, 145, 191]. Some studies reported changes in downstream performance when larger unlabelled datasets that dwarfed the available labelled examples were leveraged for pretraining [4, 108, 188]. A few even demonstrated that pretraining with unlabelled data geared for a different downstream task but that was collected using the same modality can improve downstream performance [41, 74, 99, 104]. In the extreme scenario of few-shot learning, self-supervised objectives may be employed during training [120]. The

considerable amount of evidence outlined in this review suggests that practitioners should leverage unlabelled data when available and pretrain feature extractors using SSL. The results in the subsequent chapters of this thesis include label efficiency experiments that assess the utility of pretrained backbones in cases where limited labelled examples are available.

Relative Dearth of Ultrasound Research

As depicted in Figure 2.10a, the number of papers eligible for inclusion in this review concerning ultrasound data is less than half of the number included for X-ray, CT, or MRI. Hence, there exists a need for (1) more investigations that quantify the impact of preexisting SSL pretraining tasks for ultrasound tasks and (2) studies that modify preexisting approaches or propose novel SSL methods that are suited to the ultrasound modality. Ultrasound presents additional challenges for machine learning systems compared to the other modalities, such as increased noise, the temporal dimension, acquisition-related differences in probe movement and orientation, motion artefacts, and geometrical differences across probe types and manufacturers. As a result, further work is warranted in determining the types and aspects of pretext tasks suitable for ultrasound. The studies in this thesis address this problem by focusing on SSL's use in lung ultrasound.

Theoretical Support for Empirically Validated Methods

The majority of the studies presented in the literature review used SSL methods that were presented as task-specific, instead of applying preexisting methods backed by empirical and theoretical evidence to new problems. Some deviate wildly from previous work [25, 73, 139, 197], and others are incremental changes to previously explored pretext tasks [77, 111]. The pretext tasks put forth in such studies are often fashioned with clinical and/or background knowledge about the downstream task but are mostly justified by intuition. The arguments for further use of the proposed methods typically consists entirely of empirical validation. Multiple such studies boast superior performance of their methods but do not establish statistically significant improvements [99, 192, 196].

As discussed in Section 2.3.4, some SSL methods have received theoretical justification in terms of performance on downstream task. Many studies discussed in this chapter employed such methods, such as SimCLR [29] and Barlow Twins [180]. These methods are guaranteed to improve performance on downstream tasks as long as the labels for positive pairs would be the same in the downstream task. For example, Fernandez-Quilez *et al.* [46]

employed SimCLR with a modified transformation distribution that captured differences between positive pairs that would not constitute a change of label. Azizi *et al.* [9] also employed SimCLR but expanded the pairwise relationship to include multiple acquired views of the same pathology. Applying custom data augmentation transformations that do not change the label distribution in the downstream task or defining the pairwise relationship based on preexisting clinical knowledge are viable strategies for the successful application of theoretically justified joint embedding SSL methods. Such clinical knowledge may come "for free" in that it does not require further labelling — practitioners could consider sources such as multiview examinations, multimodal studies, accompanying radiology reports, and DICOM tags. Future methods should strive to apply theoretically justified approaches to SSL pretraining where possible; otherwise, statistical significance testing should be conducted when claims are made regarding the superiority of novel methods. The studies in this thesis leverage theoretically backed SSL methods.

Comparable and Reproducible Benchmarks

A longstanding problem in machine learning for medical imaging is the lack of public datasets, which thwarts replicability of results. A considerable number of studies in this review presenting novel SSL methods for radiological imaging tasks conducted their evaluations on private datasets only. As a result, many of the results presented are not directly comparable. This review was only able to directly compare studies for a limited set of downstream tasks where authors reported performance on public datasets. Authors suggesting novel SSL methods are encouraged to evaluate their methods using public datasets, or to include results on public datasets in addition to their private datasets (e.g., [37, 163]). When evaluating on public datasets, researchers should use train/test splits that are identical to preceding studies. Furthermore, authors should endeavour to utilize identical pretraining and training sets when evaluating their approach on standard public datasets. To promote usage of public benchmarks in future studies, Table 2.1 details all public ultrasound datasets referenced in this review, providing URLs for access. Although the research presented in this thesis uses private datasets, methods are also evaluated on public lung ultrasound datasets.

The Impact of Pretraining on Generalizability

Machine learning models trained for tasks involving radiological images are utterly susceptible to performance drops under distributional shift [159]. Biases can be introduced by the distribution of confounding or mediating variables in the training set, such as labelling

discrepancies, patient demographics, acquisition technique, and device manufacturer. External validation is therefore a pivotal pre-deployment step. Multiple studies in the survey reported improvement in performance on external test sets when self-supervised pretraining was conducted [8, 9, 150, 195], but further work is required to confidently characterize this phenomenon. This thesis addresses generalizability by evaluating self-supervised models on test sets that originate from different healthcare institutions than those comprising training data.

Name [Citation]	Description	Images	Videos	Patients
BreastMNIST [172]	Identical to BUSI [1]. Part of MedMNIST.	780	-	600
BUI [131]	Breast ultrasound images labelled as containing benign or malignant tumours.	250	-	233
BUSI [1]	Breast ultrasound images with classification labels for normal, benign lesion, and malignant lesions.	780	-	600
DDTI [125]	Thyroid ultrasound images with segmentation masks for benign and malignant lesions, diagnostic tags, and TI-RADS values [148]	347	-	299
CAMUS [87]	Echocardiograms labelled for segmentation and volume estimation.	-	500	500
EchoNet-Dynamic [121]	Echocardiography videos with end diastolic and end systolic volume labels	-	10 030	10 030
POCOVID-Net [19]	Links to lung ultrasound videos labelled for COVID-19, other viral pneumonia, bacterial pneumonia, and healthy lung.	1103	64	-
TN-SCUI2020 [141]	Thyroid ultrasound videos with segmentation labels for thyroid nodules	-	3644	3644
UDIAT [175]	Breast ultrasound images with segmentation masks corresponding to tumours.	163	-	163
US-MTD120K [101]	Thyroid, breast, liver, and fetal ultrasound images, originating from two distinct sites. Multiclass labels are provided for each type of examination. The majority of the images are duplications that have been transformed using either rotation, horizontal reflection, or Gaussian blur.	120 354	-	3600
UXTD [44]	Tongue ultrasound videos recorded while subjects were speaking in response to a diverse set of prompts. Several types of speech annotations are provided.	-	- 58	

Table 2.1: The public ultrasound datasets referenced in this Section 2.2.2, including links to request or download the data.

Chapter 3

Self-Supervised Pretraining for Multiple Tasks in B-Mode Lung Ultrasound

3.1 Introduction

As outlined in Chapter 1, lung ultrasound (LU) is a point-of-care ultrasound (POCUS) examination that is performed in acute care settings to rapidly narrow down differential diagnoses for patients in respiratory distress. In addition to its enhanced portability, safety, and affordability, LU has exhibited diagnostic accuracy for a variety of respiratory conditions that is comparable to traditional modalities, such as chest radiography [2, 91, 116, 169] and computed tomography [34, 116, 117, 164]. Despite mounting evidence for its efficacy, there are barriers to the widespread adoption of POCUS, including reduced availability of training and lack of access to devices [21, 54, 58].

The literature survey in Chapter 2 provided a wealth of evidence in favour of self-supervised pretraining for ultrasound imaging problems, particularly when labelled examples are not abundant. Indeed, a major barrier to the development of machine learning models for tasks in POCUS is lack of access to curated, labelled datasets [102]. In addition to the sparsity of LU expertise, the expense of soliciting experts to manually label retrospectively acquired ultrasound videos is prohibitive. As a result, there is remarkable value in discovering techniques that can reduce the amount of labelling required for retrospectively collected datasets. The primary objective of the research in this chapter was to determine if contemporary self-supervised learning (SSL) methods are a viable solution for

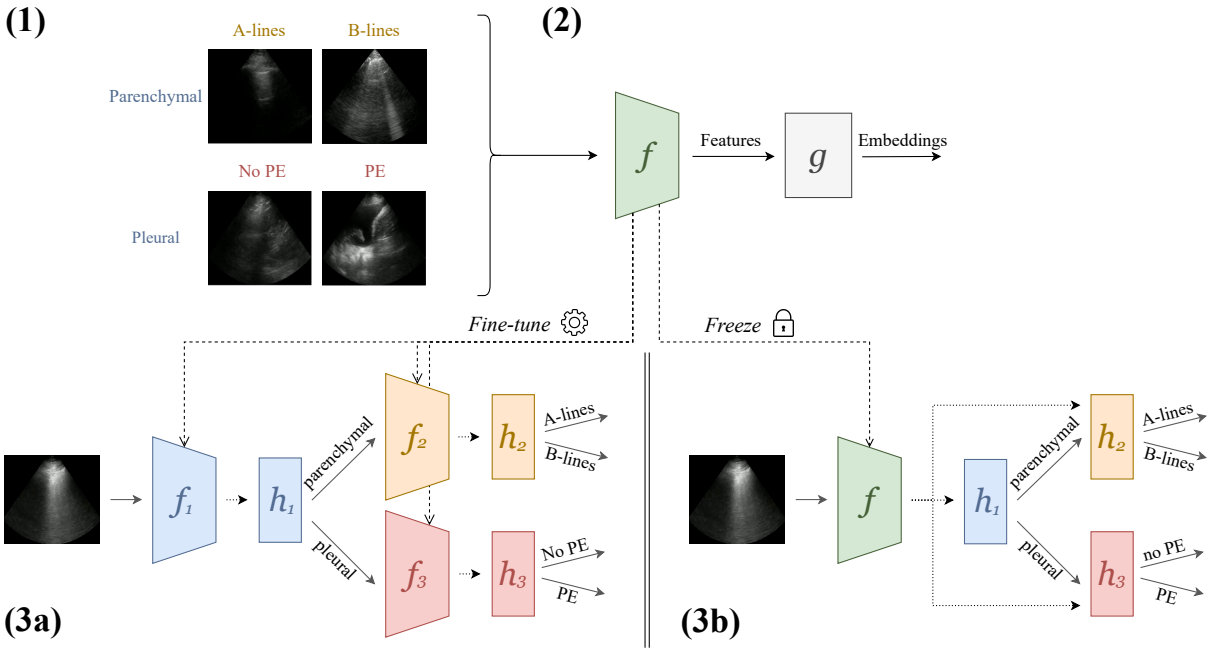


Figure 3.1: An overview of the methods described in this chapter. **(1)** Three tasks were identified for lung ultrasound (LU) image classification: parenchymal versus pleural views, A-lines versus B-lines (applicable to parenchymal views), and pleural effusion (PE) versus no pleural effusion (applicable to pleural views). **(2)** A convolutional feature extractor f was pretrained to minimize a self-supervised objective, using unlabelled and labelled LU images as input and trainable projector g. **(3a)** Task-specific models were defined by appending linear classifier or multilayer perceptron h_i to copies of pretrained f. The models were trained end-to-end for each task using labelled data. **(3b)** An alternative framework in which f's weights were not fine-tuned. Instead, task-specific models h_i were trained that each received f's feature representations as input.

improving performance in LU classification tasks, particularly when there are few labelled examples in a retrospectively acquired LU dataset. We considered multiple tasks to ensure that any observed benefits of SSL were not confined to one particular LU task.

The secondary objective of this chapter was to explore whether SSL methods can produce a single set of feature representations that are useful across different LU tasks. More specifically, we wished to determine if such representations could be useful for the hierarchical arrangement of LU interpretation tasks. LU interpretation involves the recognition of multiple artefacts that narrow differential diagnoses in emergency and critical care sce-

narios, hereafter referred to as *multi-task LU interpretation*. As is common across other medical imaging modalities, LU interpretation can be conducted in a hierarchical manner. Interpreters engage in the predictive process of a decision tree, beginning with the root node and traversing down a single path, guided by decisions at each node. Examples of hierarchical interpretation from other modalities include the distinction of malignant pulmonary nodules on CT [109] and the identification of lipomatous tumours on MRI [140]. Past work in machine learning-based hierarchical medical imaging classification has resorted to training entirely separate classifiers for each node in the tree [176]. Our study sought to determine if a single feature extractor could produce meaningful representations for multi-task LU interpretation. We hypothesized that self-supervised pretraining is suited for the task of developing a feature extractor that is useful for multiple classification tasks. The weights of the feature extractor could be fine-tuned for individual subsequent tasks (Figure 3.1, 3a). Alternatively, holding the weights constant would facilitate the addition of new tasks to the multi-task LU interpretation system by training classifiers on top of the representations (Figure 3.1, 3b).

The core contributions of this chapter are as follows:

- An investigation of the suitability of self-supervised feature extractors for multi-task interpretation of B-mode lung ultrasound
- A tree-based classification strategy in which the inputs to the root node are obtained from a feature extractor pretrained with self-supervised learning

Section 3.2 provides a focused background of the SSL methods investigated in this chapter and a summary of prior approaches to multi-task medical image interpretation. Section 3.3 describes the LU interpretation tasks, datasets, and SSL pretraining protocol employed in this chapter. Section 3.4 details the experimental evaluation. It also gives runtime statistics for each task and compares the fine-tuning of end-to-end models for each task against the training of MLPs on representations outputted by a single pretrained backbone. Lastly, conclusions and recommendations for machine learning practitioners are given in Section 3.5.

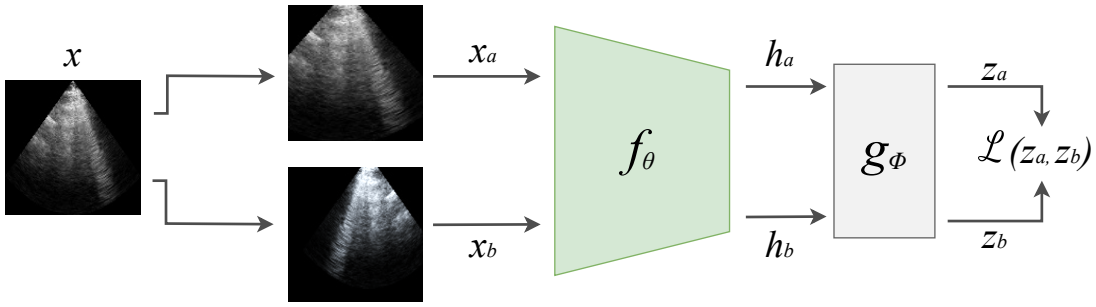


Figure 3.2: A depiction of the forward pass for a batch of images in a standard joint embedding SSL task. Batches of images are subjected to stochastic data transformations twice, producing distorted views x_a and x_b, which are passed through the backbone f_θ to yield feature representations h_a and h_b. The projector g_ϕ transforms h_a and h_b into embeddings z_a and z_b, respectively. Typically, the objective function $\mathcal{L}(z_a, z_b)$ is optimized to maximize the similarity of z_a and z_b.

3.2 Background

3.2.1 Joint Embedding Self-Supervised Learning

Broadly, SSL is a form of unsupervised representation learning that is employed to pretrain a feature extractor for transfer learning. The weights of the pretrained feature extractor can then be used to initialize a new model trained to solve a supervised learning task using a labelled dataset. In the joint embedding framework of SSL, semantically related *positive pairs* of images are customarily passed through the feature extractor, with the output being sent through a projection head (typically an MLP), producing embeddings. A common approach for defining positive pairs is to produce two distinct distortions of the same image, ensuring that the distorted views do not alter the semantic content of the image. Refer to Figure 3.2 for a visual depiction of a prototypical joint embedding method. Further background details on joint embedding SSL are available in Section 2.3.3.

For an account of prior uses of SSL in B-mode ultrasound, please consult Section 2.4. This chapter adds to previous literature exploring SSL for LU tasks by (1) investigating the utility of the same pretrained feature extractor for multiple LU tasks and (2) including experimentation with non-contrastive SSL methods.

3.2.2 Multi-Task Medical Image Interpretation

Several studies have addressed multi-task learning for multi-task medical imaging interpretation. For instance, Zhang *et al.* [187] trained a single neural network with dedicated output layers for the classification of carotid plaques and estimation of the degree of stenosis on CT angiography imaging. Xu *et al.* [171] proposed a single CNN architecture for abdominal ultrasound view classification and landmark localization using feature representations from intermediate residual blocks as input for both tasks. Focusing on hierarchical interpretation, Fu *et al.* [49] proposed a system for medical image classification consisting of a CNN followed by a decision tree in which each node is a linear classifier [49]. Decision trees with neural network nodes have also been proposed [176].

In this chapter, we show that a single CNN pretrained with self supervision provides sufficient feature representations for multiple tasks, including tasks arranged hierarchically. Note that the methodology in this work is distinct from multi-task learning in that it explores the feasibility of reusing a single self-supervised pretrained feature extractor for the development of multiple LU classifiers.

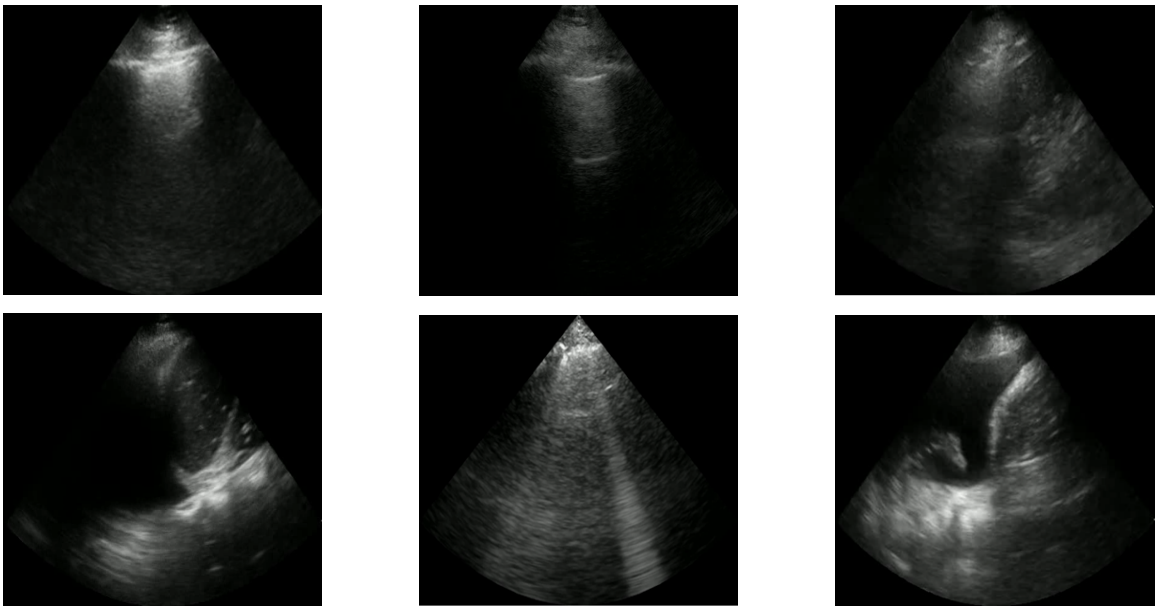
3.3 Methods

3.3.1 Lung Ultrasound Classification Tasks

The LU interpretative workflow addressed in this work has been described as a decision tree [143]. After determining the view, the interpreter traverses down the tree to look for increasingly specific artefacts that refine a differential diagnosis. We focused on three binary classification tasks for LU image interpretation: view classification (`View`), A-line versus B-line classification (`AB`), and pleural effusion detection (`PE`). The `AB` task is applicable to parenchymal views on LU, and the `PE` task is applicable to pleural views on LU. Table 3.1 summarizes these tasks, and Figure 3.1 displays emblematic examples from each class. Section 2.2 in the previous chapter provides greater detail on these LU tasks. To our knowledge, this chapter constitutes the first effort to study the impact of self-supervised pretraining for the `View`, `AB`, and `PE` LU tasks.

3.3.2 Datasets

Datasets from one local and one external healthcare institution were extracted from a private repository of LU videos. Appendix A details how access to the private repository



(a) `View`: Parenchymal (top), pleural (bottom)

(b) `AB`: A-lines (top), B-lines (bottom)

(c) `PE`: No PE (top), PE (bottom)

Figure 3.3: Examples of each class for each LU binary classification task: `View` (a), `AB` (b), and `PE` (c).

Task	Description	Negative Class	Positive Class
`View`	Distinguishing the type of LU view	Parenchymal	Pleural
`AB`	Normal versus abnormal lung tissue, as indicated by the presence of A-lines and B-lines, respectively. Visualized in parenchymal views.	A-lines	B-lines
`PE`	Absence or presence of pleural effusion. Visualized in pleural views.	No effusion	Effusion

Table 3.1: A summary of the lung ultrasound tasks addressed in this chapter.

was facilitated. The dataset had been previously labelled for the `View`, `AB`, and `PE` tasks by competent LU interpreters during prior work [6, 152]. The labelled portion of the local dataset was split by anonymous patient identifier into a training set (70%), validation set (15%), and test set (15%), and the external dataset was reserved for testing only.

	Local				External
	Unlabelled	Train	Validation	Test	Test
Patients	6164	1540	330	329	
Videos	20 000	4130	859	939	
Patients	4919	975	210	210	53
Images	2 192 361	757 492	161 302	157 797	45 317
`View`	-	520 128 / 237 364	113 436 / 47 866	109 684 / 48 113	33 483 / 11 834
`AB`	-	195 000 / 88 361	43 980 / 17 269	43 378 / 17 719	12 173 / 12 078
`PE`	-	110 165 / 40 119	27 624 / 8236	21 081 / 9838	6830 / 2641

Table 3.2: Breakdown of the institutional ultrasound datasets used in this chapter. For each LU binary classification task, $x \,/\, y$ indicates the number of negative and positive examples, respectively.

Local videos with no labels were used only during self-supervised pretraining. Table 3.2 details the cardinalities and class distribution of these datasets. Regions peripheral to the ultrasound beam were expunged of extraneous visual artefacts, and the images were cropped to the boundaries of the beam. All images were downsampled to 128×128 pixels, which is consistent with prior work for similar tasks [6, 152].

We also evaluated the efficacy of SimCLR-pretrained weights on the public COVIDx-US dataset [43], splitting it by video identifier into a training, validation, and test set. Although patient identifiers were not available for every video, we ensured that multiple videos from the same patient identifier were contained in the same set. COVIDx-US contains 243 LU videos (29 651 images) originating from a variety of manufacturers and clinical sources. Each example belongs to one of four classes: normal lung, COVID-19, non-COVID-19 pneumonia, and a general class encompassing other pathologies. The dataset was sourced from openly available LU examinations acquired at a variety of institutions with an assortment of ultrasound devices.

3.3.3 Self-Supervised Pretraining

Three joint embedding SSL methods were trialled to produce pretrained models for each LU task: SimCLR (with $\tau = 0.1$) [29], Barlow Twins (with $\lambda = 0.005$) [180], and VICReg (with $\lambda = 25$, $\mu = 25$, $\nu = 1$) [12]. As was done in the original studies, positive pairs were produced by distorting images by applying stochastic data augmentations sampled from a family of transformations. Figure 3.4 provides examples of augmented views of B-mode

images from the local dataset. Below is the list of transformations, where P indicates the probability of a transformation being applied:

1. Random square crop with width $c \sim \mathcal{U}(0.5, 1.0)$ of the image's width. $(P = 0.8)$.
2. Horizontal flip. $(P = 0.5)$
3. Gaussian blur with a kernel of 5×5 pixels and $\sigma \sim \mathcal{U}(0.1, 2.0)$. $(P = 0.2)$
4. Multiplicative Gaussian noise, with SD $\sigma \sim \mathcal{U}(0.0, 0.1)$. $(P = 0.5)$.
5. Brightness adjustment by $c \sim \mathcal{U}(0.5, 1.5)$. $(P = 0.7)$.
6. Contrast adjustment by $c \sim \mathcal{U}(0.5, 1.5)$. $(P = 0.7)$. With probability 0.5, this occurs before brightness adjustment.

Feature extractors were pretrained for 15 epochs using the union of the unlabelled and training images. The MobileNetV3Small [71] architecture, initialized with ImageNet-pretrained weights, was employed as the backbone for all pretraining. The architecture choice was motivated by its efficacy and inference speed when implemented on mobile devices, which is conferred by design elements such as inverted residual blocks, and squeeze-and-excitation blocks [71]. The output of MobileNetV3Small is a 576-dimensional feature representation vector.

3.3.4 Evaluation Protocol

We compared pretrained models with fully supervised models initialized with ImageNet-pretrained weights. The following experiments were conducted to determine the pretrained backbones' effectiveness at learning the LU tasks.

- **Linear classification (LC)**: The weights of the feature extractor were held constant, and a linear classifier was trained using its outputted feature representations.
- **Fine-tuning (FT)**: The weights of both the feature extractor and a linear head were trained.
- **Nonlinear classification (NC)**: The weights of the feature extractor were held constant, and a nonlinear head was trained on the representations. The head consisted of a MLP with a single hidden layer of 32 nodes with rectified linear unit activation.

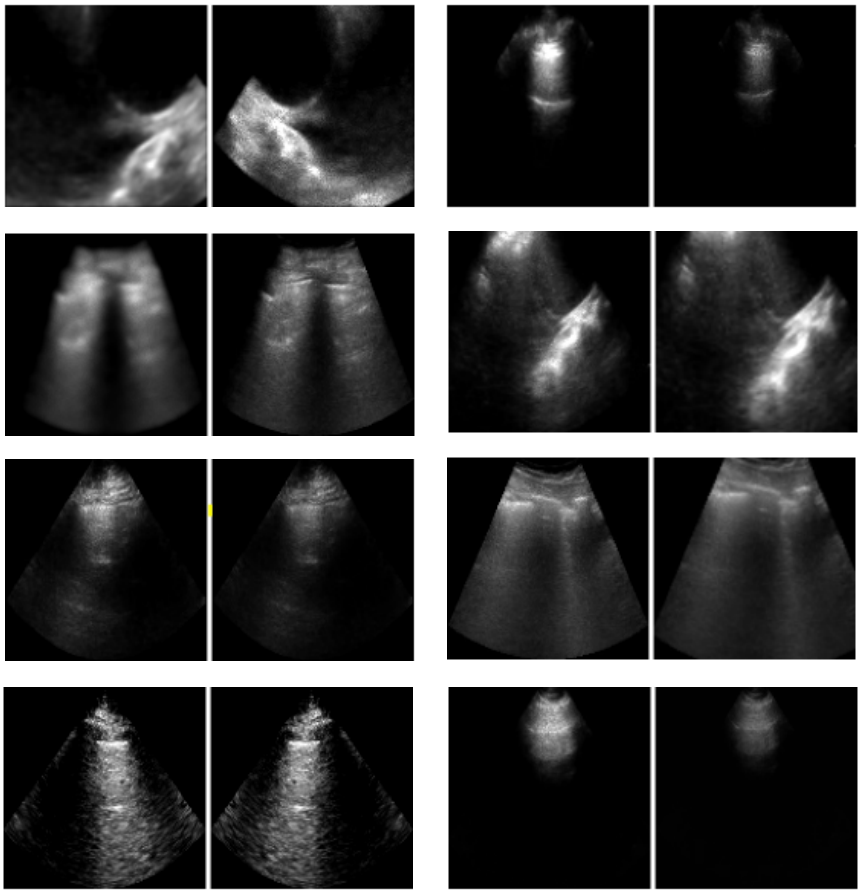


Figure 3.4: Examples of positive pairs of B-mode images produced by applying data augmentation to the same image twice.

Figure 3.1 (parts 3a & 3b) illustrates how FT and NC each implement hierarchical LU interpretation for the tasks of interest. In all trials, the initial learning rates for the feature extractor and head were 1×10^{-5} and 1×10^{-4}, respectively. The learning rates were multiplied by a factor of $e^{-0.02}$ each epoch. Models were trained for 10 epochs to minimize the binary cross-entropy loss function. The weights resulting in the lowest validation set loss were retained. We assessed model performance by determining the area under the receiver operating characteristic curve (AUC) on the local and external test sets. All experiments were conducted using a system with an Intel i9-10900K CPU at 3.7 GHz and a Nvidia GeForce RTX 3090 GPU.

3.4 Results

Task	Pretraining	Local test set			External test set		
		LC	FT	NC	LC	FT	NC
`View`	SimCLR	0.966	0.982	0.976	0.887	0.922	0.910
	Barlow Twins	0.959	0.978	0.973	0.889	0.920	0.908
	VICReg	0.960	0.980	0.975	0.894	0.917	0.909
	None	0.951	0.978	0.976	0.908	0.925	0.903
`AB`	SimCLR	0.949	0.976	0.969	0.890	0.896	0.898
	Barlow Twins	0.940	0.958	0.948	0.903	0.886	0.894
	VICReg	0.934	0.957	0.940	0.880	0.904	0.902
	None	0.923	0.945	0.939	0.820	0.833	0.856
`PE`	SimCLR	0.832	0.925	0.894	0.929	0.941	0.928
	Barlow Twins	0.855	0.906	0.893	0.845	0.940	0.923
	VICReg	0.849	0.935	0.871	0.878	0.936	0.921
	None	0.857	0.867	0.946	0.854	0.821	0.972
Mean	SimCLR	0.916 [6.0]	**0.961 [2.6]**	0.946 [3.7]	**0.902 [1.9]**	**0.920 [1.8]**	**0.912 [1.2]**
	Barlow Twins	**0.918 [4.5]**	0.947 [3.0]	0.938 [3.3]	0.879 [2.5]	0.915 [2.2]	0.908 [1.2]
	VICReg	0.914 [4.7]	0.957 [1.8]	0.929 [4.3]	0.884 [0.7]	0.919 [1.3]	0.911 [0.8]
	None	0.910 [3.9]	0.930 [4.7]	**0.954 [1.6]**	0.861 [3.6]	0.860 [4.6]	0.910 [4.8]

Table 3.3: AUC evaluated on the local and external test sets for the linear classification (LC), fine-tuning (FT), and nonlinear classification (NC) experiments. Results are presented for each of the `View`, `AB`, and `PE` tasks. The bottom row reports the mean [SD $\times 10^{-2}$] across tasks, with bold typeface indicating the best-performing pretraining strategy.

3.4.1 Linear Classifier Evaluation (LC)

Feature extractors were pretrained using SimCLR [29], Barlow Twins [180], and VICReg [12]. To evaluate the separability of the resulting representations with respect to the three LU tasks, linear classifiers were trained on the output of the feature extractors. In each trial, the feature extractor's weights were held constant, and a perceptron was fitted for each binary LU task, using the pretrained representations as input. Table 3.3 provides the performance of classifiers trained for the LC experiment on the local and external test sets. AUC was designated as the primary evaluation metric, but additional classification metrics are reported in Appendix B.

We compared the test AUC exhibited by classifiers initialized with self-supervised pretrained weights against classifiers initialized with ImageNet-pretrained weights (i.e., no self-supervision). Hereafter, we refer to the models initialized without self-supervised pretraining as *fully supervised*. In the case of linear evaluation, self-supervised pretraining resulted in greater performance on local test data for `View` and `AB`, but not for `PE`. On the external test set, the linear classifiers for pretrained models outperformed fully supervised models on `AB`, but not for `View`. The lack of difference for `View` may be the result of the greatly increased number of training examples labelled for `View` compared to the other tasks. SimCLR- and VICReg-pretrained models performed better on the `PE` task with external test data than fully supervised models, but Barlow Twins-pretrained models did not — a finding that was unique to this experiment.

3.4.2 Fine-tuning Evaluation (FT)

As in the LC experiment, a single-node fully connected layer was appended to the pretrained feature extractors. The entire network was then fine-tuned, facilitating task specialization by the feature extractor. The learning rate for the feature extractor was set to a tenth of that of the output layer (specifics are provided in Section 3.3.4). Table 3.3 provides the AUC evaluated on the local and external test sets for the FT experiment. Each pretraining method achieved greater AUC on the local test set across all tasks. On the external test set, self-supervised pretrained models resulted in a minimum AUC improvement across pretraining methods of 0.053 and 0.115 for `AB` and `PE`, respectively. Improved external test set performance suggested improved generalizability of the pretrained classifiers. On the external `View` test set, fully supervised methods achieved the greatest AUC by a very small margin. In fact, the performance of fully supervised and pretrained models was close for `View` on both local and external test data – a finding that may be due to the substantially larger number of labelled examples available for the `View` task compared to the others. Section 3.4.5 explores how these results differ when considerably fewer labels are available for each task.

The general result that fine-tuned models initialized with self-supervised pretrained feature extractors outperform fully supervised baselines is consistent with previous studies that investigated the utility of ultrasound-specific contrastive pretraining objectives for COVID-19 classification [13, 33]. A caveat is that the experiments by Chen *et al.* [33] comparing pretrained and fully supervised models were conducted in a semi-supervised setting, as opposed to self-supervision. This chapter expands on previous results, showing that (1) self-supervised pretraining improved performance of fine-tuned classifiers on three

additional LU interpretation tasks, and (2) that non-contrastive self-supervised approaches also improved LU classifiers for these tasks.

3.4.3 Nonlinear Classification (NC)

The last experiment type performed on local and external test was nonlinear classification. Similar to the LC experiment, the objective was to evaluate the ability of unmodified pretrained feature representations to serve as inputs to a simple classifier. Unlike LC, NC experiments make use of a 2-layer MLP, facilitating nonlinear decision boundaries. It was expected that such models would not perform as well as fine-tuning, since the feature extractor was barred from specializing in the task. However, should the performance gap with lightweight nonlinear classifiers be acceptably small, a hierarchical multi-task interpretation system could be constructed that reuses representations computed with one pass of the pretrained model. Here we detail the performance of lightweight classifiers on test data and compare them to fine-tuned models.

As shown in Table 3.3, pretraining did not consistently result in clear improvement on local test data. Similar to FT, there was little difference in performance on `View`. On the `AB` task, multiple pretrained models outperformed their fully supervised counterparts on local and external test data. Most notably, a fully supervised MLP achieved higher AUC on the `PE` task on the local and external test sets than any of the pretrained models. While we are not able to offer a direct explanation for this finding, we speculate that the performance gap was related to the fact that there may have been considerably fewer examples of pleural effusions in the pretraining data. With 27.0% of the images with a `PE` label containing a pleural effusion and 31.3% of the images in the entire training set labelled as being a pleural view, we estimate that the approximate prevalence of pleural effusions across the dataset was 8.3%. It is possible that the pretrained models learned to produce stronger representation for parenchymal views and normal pleural views.

Experiment	SimCLR [local]	SimCLR [COVIDx-US]	Supervised [ImageNet]
FT	0.627	**0.709**	0.623
NC	0.621	**0.751**	0.685

Table 3.4: Mean class-wise AUC on the COVIDx-US test set for FT and NC.

3.4.4 Public Dataset Evaluation

To promote experimental replicability, we investigated the effect of self-supervised pretraining with SimCLR on COVIDx-US, a public LU dataset [43]. As shown in Table 3.4, models pretrained with SimCLR on the COVIDx-US training set achieved better mean class-wise test AUC than fully supervised models. These results were consistent with the improvement in COVID-19 pneumonia classification observed by previous works employing contrastive self-supervised pretraining [13, 33]. To explore the transferability of pretrained weights, we conducted a separate training run using weights pretrained with SimCLR on the local LU dataset. Although COVIDx-US contained less than a tenth of the number of videos in the local training set alone, it was amalgamated from a variety of institutions and device manufacturers. Despite having been pretrained on several more videos, the models pretrained on local data greatly underperformed those pretrained on COVIDx-US alone, while performing comparably to fully supervised models. The results highlighted the importance of pretraining on a data distribution that is similar to the LU interpretation task of interest.

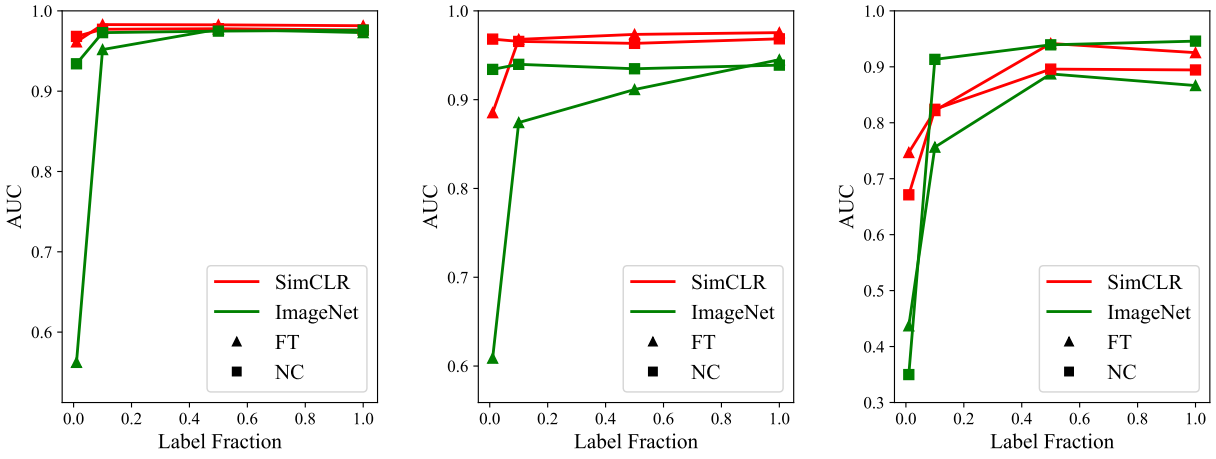
3.4.5 Label Efficiency

FT and NC were repeated for ImageNet-pretrained and SimCLR initialization using 1%, 10%, and 50% of the training set to evaluate the label efficiency of self-supervised pretrained models. As depicted in Figure 3.5, self-supervision improved performance on the local test set in most cases. Moreover, the performance gain realized with SimCLR pretraining was largest when training with 1% of the labels. Recalling from Sections 3.4.1, 3.4.2, and 3.4.3 that pretrained models did not clearly outperform fully supervised models for the `View` task, the label efficiency experiments highlighted the utility of self-supervised pretraining for `View` when fewer labelled training examples were available.

A notable finding was the leading performance of the fully supervised MLPs for `PE` (the NC experiment), which we suspect was related to the smaller size of the local test set. However, fully supervised MLPs trained on only 1% of the available `PE` labels greatly underperformed those with feature representations from the SimCLR-pretrained model.

3.4.6 Qualitative Evaluation of Representations

Seeking to better understand the results, we visualized two-dimensional t-SNE [110] projections of the representations outputted by both an ImageNet-pretrained and a SimCLR-pretrained feature extractor. As can be seen in Figure 3.6, the projections for `PE` were



(a) `View` (b) `AB` (c) `PE`

Figure 3.5: Local test AUC for supervised models initialized with ImageNet-pretrained weights and SimCLR-pretrained weights. Results are provided for the fine-tuning (FT) and nonlinear classification (NC) experiments training on various fractions of the labelled dataset.

not well-separated, even after pretraining with SimCLR, offering insight into the comparatively diminished improvement imbued by pretraining. In contrast, the projections suggested that self-supervised pretraining improved the separability of the data for the `AB` task, which was reflected in the decidedly stronger performance of the SimCLR-pretrained model. The difference in performance after self-supervised pretraining was less clear for `View`, which may have occurred because there were significantly more labelled examples available for `View` (see Table 3.2). Moreover, the t-SNE projections for `View` exhibited separability before and after SimCLR pretraining.

3.4.7 Inference Efficiency

Recall that LU interpretation consists of a hierarchically arranged set of tasks. Instead of training several task-specific models, the representations outputted by one self-supervised LU feature extractor could be useful for multiple such tasks. Since speed is one of the essential qualities of POCUS, the design of automated LU diagnostic assistive software should prioritize runtime minimization. Real-time device inference could be accomplished by using copies of the output of a single feature extractor as input to multiple lightweight

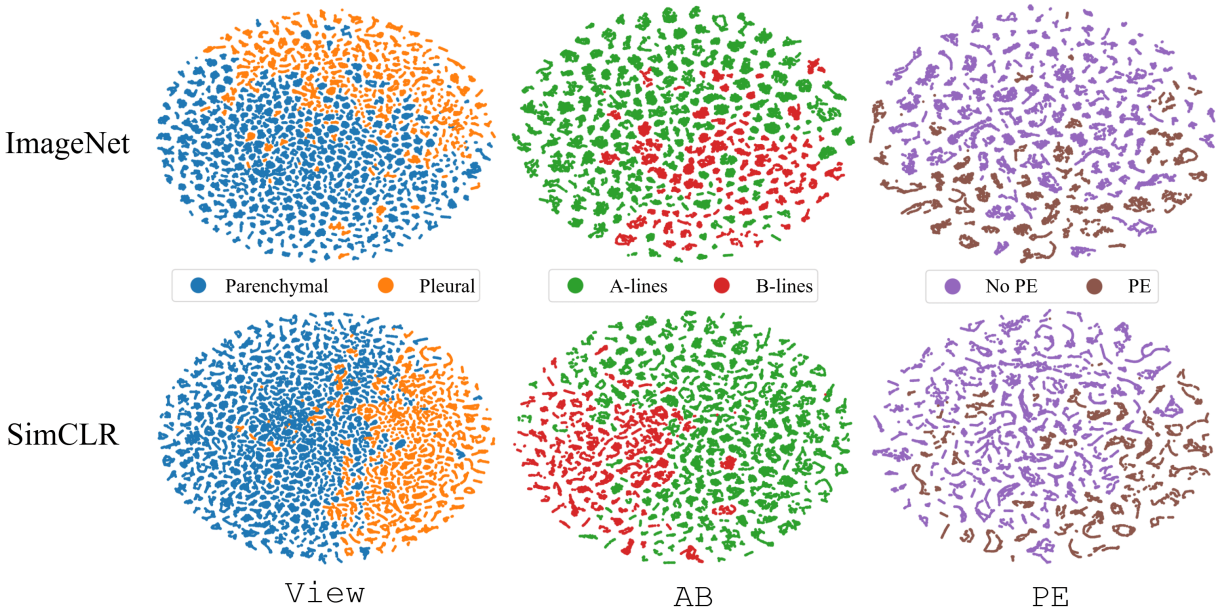


Figure 3.6: A comparison of t-SNE projections of representations for the examples in the local test set outputted by a feature extractor initialized with ImageNet-pretrained weights before SimCLR pretraining (top) and after SimCLR pretraining (bottom).

MLP classifiers. Furthermore, as LU is a rapidly evolving diagnostic tool, classifiers for novel tasks could be integrated by training lightweight classifiers on the pretrained model's representations.

Although the results indicated that task-specific fine-tuning yielded the greatest per-task performance, any future work that improves nonlinear classification with pretrained weights would improve the execution time of a hierarchically arranged end-to-end LU interpretation workflow. To quantify the potential runtime gains of using a single feature extractor for multi-task LU classification, we compared the prediction time of two serially arranged fine-tuned CNNs (Figure 3.1, 3a) against one feature extractor and two subsequent MLP classifiers (Figure 3.1, 3b). Both configurations reflect the decision tree connecting the `View`, `AB`, and `PE` tasks in the simplified LU interpretation workflow. After conducting 1000 serial predictions, the former took an average of 0.116 s (SD 0.003 s), while the latter took an average of 0.059 s (SD 0.001 s), underlining the runtime advantage of multi-task inference with a shared feature extractor. With each feature extractor and MLP requiring 3.7×10^7 and 3.7×10^4 floating-point operations respectively, reusing the output of a single feature extractor as input to multiple task-specific MLPs would save con-

siderable computational resources. The LU diagnostic tree depicted in Figure 3.1 would require approximately half the floating-point operations if each node was a lightweight MLP instead of an entire CNN. Future work should focus on improving the applicability of unaltered feature representations from self-supervised pretrained models for multiple LU classification tasks.

3.5 Discussion

In this chapter, joint embedding SSL methods were observed to improve the performance of classifiers on a variety of basic LU tasks, particularly when only a small fraction of labels were available. These findings are consistent with the literature reviewed in Chapter 2 that SSL concerning the effectiveness of SSL in other ultrasound examinations. Fine-tuning self-supervised pretrained models for each task consistently yielded the greatest performance gains for each task, with SimCLR-pretrained models improving across-tasks average AUC improvement of 0.032 and 0.061 on local and external test sets, respectively. When holding the weights of pretrained feature extractors constant, linear classifiers trained on representations from self-supervised models consistently achieved greater across-task average AUC on local and external test data. MLP classifiers trained on feature representations outputted by self-supervised pretrained models did not outperform fully supervised models on all tasks. Nevertheless, low-dimensional projections of feature representations provided qualitative evidence that the test examples were well-separated with respect to two of the three tasks studied.

Based on the results of this chapter, machine learning practitioners working on automated LU interpretation software should strongly consider pretraining a feature extractor using any of the image-based contrastive or non-contrastive SSL methods investigated in the experiments, particularly when the majority of the available images are unlabelled. Given that pretraining improved the performance of fine-tuned models for multiple LU tasks, the results support the conclusion that self-supervised pretraining is a viable method to boost LU classifier performance. Moreover, developers may benefit from reduced dependence on expensive labelling expertise, achieving high classifier performance even without an abundance of labels. Despite the benefits of fine-tuning, it was observed that training two-layer MLP networks on the feature representations outputted by pretrained models did not consistently improve performance across tasks. As such, practitioners should consider fine-tuning for specific LU tasks instead of training shallow MLPs on a frozen pretrained backbone's representations. Future work could study whether deeper MLPs are required to match the performance of fine-tuned backbones.

In summary, this chapter demonstrated that joint embedding self-supervised pretraining is a practical strategy for improving performance on B-mode LU classification tasks when a fraction of the available data is labelled. More broadly, the findings imply that access to immense clinical labelling resources is not necessary to develop proficient LU classifiers; rather, unlabelled data can be adapted via self-supervised learning to achieve improved performance. Lastly, while this chapter focused on improving performance of LU classifiers in B-mode ultrasound imaging, one of the most critical LU interpretation tasks is conducted using M-mode. In the next chapter, we will see that self-supervised pretraining also confers tangible benefits for machine learning classifiers in M-mode ultrasound.

Chapter 4

Self-Supervised Pretraining for Lung Sliding Classification in M-Mode Lung Ultrasound

4.1 Introduction

In the previous chapter, we demonstrated that self-supervised pretraining improved the performance of machine learning models trained to perform lung ultrasound (LU) classification tasks. The image inputs for each task investigated in Chapter 3 were B-mode ultrasound images. This chapter examines the use of self-supervised learning (SSL) for a critical task in M-mode LU: distinguishing between the presence and absence of the lung sliding finding. The lung sliding sign is incredibly useful in the diagnostic workup for pneumothorax (PTX) — a potentially life-threatening acute condition in which air occupies the space between the parietal pleura and the visceral pleura of the lungs, resulting in collapse of the lung. Rapid identification of PTX is crucial in emergency, critical, and acute care settings to expedite medical intervention.

As outlined in Section 1.1, LU is a quick, inexpensive, portable, imaging examination that does not expose patients to radiation. Despite its low usage compared to chest radiographs, LU has been shown to exhibit superior diagnostic performance for the diagnosis of PTX [2, 116]. The lung sliding sign, caused by the normal motion of the pleura, has been described as a means to rule out PTX [95]. On B-mode ultrasound, lung sliding manifests as dynamic shimmering of the pleural line on parenchymal views of the lung. Notably,

the presence of lung sliding excludes a diagnosis of PTX at the site where the ultrasound probe is placed [95]. Conversely, PTX is likely present when lung sliding is absent.

In addition to being appreciable on B-mode ultrasound video, the presence or absence of lung sliding can be distinguished by examining M-mode ultrasound images of scan lines that intersect the pleural line. The *seashore sign* is a previously described pattern for M-mode lung ultrasound that signals the presence of lung sliding [92, 97]. Conversely, the *barcode sign* (also known as the *stratosphere sign*) indicates that lung sliding is not present [92, 97].

Previous studies have demonstrated that deep convolutional neural networks (CNNs) can be trained to distinguish between the presence and absence of lung sliding on M-mode images [78, 158]. Prior studies were limited by the amount of labelled data available for training and evaluation. Furthermore, previous studies initialized their networks using weights pretrained on the ImageNet dataset [35]. Despite the fact that M-mode images are profoundly distinct from the natural images present in ImageNet, it is common for medical imaging studies to leverage ImageNet-pretrained weights. They are publicly available for several common architectures and are able to extract low-level features present in medical images. Unfortunately, there are no publicly available equivalents for M-mode images, let alone LU.

This chapter explores the impact of self-supervised pretraining for the downstream task of detecting absent lung sliding in M-mode LU. It builds on the evidence presented in Chapter 3 that self-supervised pretraining is beneficial for LU downstream tasks but focuses on a problem in M-mode ultrasound instead of B-mode. The experiments assess multiple SSL methods, weight initialization techniques, data augmentation strategies, and the utility of unlabelled data. Crucially, we demonstrate that incorporating large volumes of unlabelled M-mode images during the pretraining phase improves the performance of a fine-tuned classifier on external datasets. The major contributions of this chapter are as follows:

- A pairwise relationship for contrastive and non-contrastive learning that is specific to M-mode images
- A data augmentation pipeline specific to M-mode images in the context of pretraining
- A comprehensive investigation of factors that influence the utility of SSL pretraining for the downstream task of absent lung sliding detection, such as label efficiency, ImageNet initialization, and data augmentation

- Evidence that the inclusion of unlabelled data results in improved generalization to external datasets for absent lung sliding detection

Figure 4.1 summarizes the methods. To the best of our knowledge, this chapter constitutes the first attempt to investigate the efficacy of SSL for M-mode ultrasound tasks, even outside of the purview of LU.

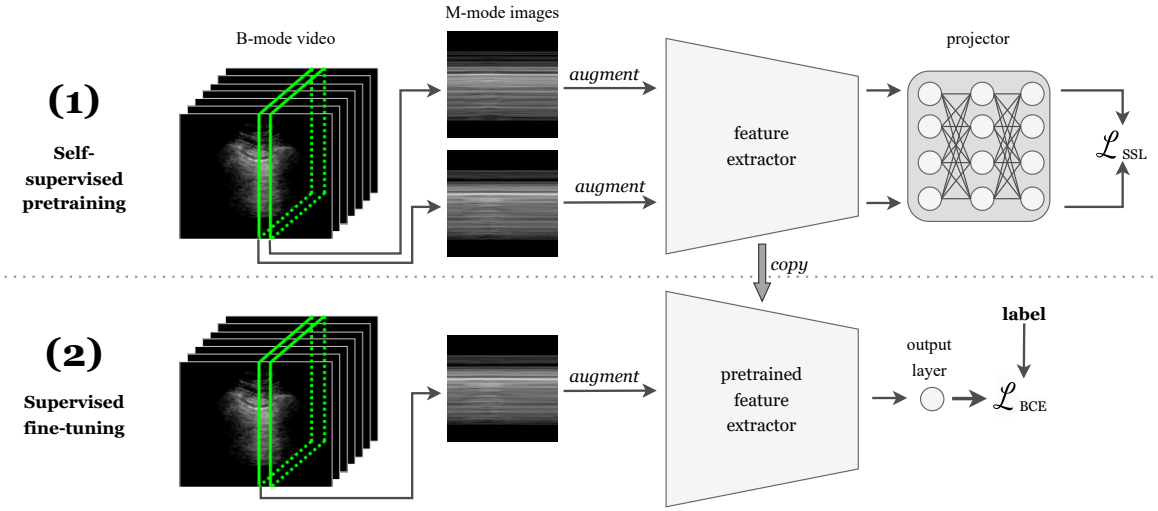


Figure 4.1: An overview of the methods in this chapter. (1) Self-supervised pretraining using pairs of M-mode images extracted from the same B-mode LU video. Both M-mode images are passed through a joint embedding SSL architecture, consisting of a CNN feature extractor and MLP projector in series. The objective, $\mathcal{L}_{SSL}$, is computed using the embeddings outputted by the projector. The pretrained CNN feature extractor is retained and the MLP projector is discarded. (2) A single-node output layer is appended to the pretrained feature extractor. The resulting model is fine-tuned to solve the downstream task of absent lung sliding detection by minimizing the binary cross entropy loss with respect to the labelled training set.

4.2 Background

Prior works have demonstrated the utility of self-supervised pretraining in B-mode medical ultrasound. Section 2.4 discussed several applications of self-supervised learning to B-

mode medical ultrasound interpretation tasks. Chapter 3 studied the impact of SSL on B-mode LU downstream tasks and found that pretrained backbones outperformed backbones initialized with random weights or ImageNet-pretrained weights.

In this section, we provide additional background on the use of machine learning in lung sliding classification for M-mode LU and discuss the importance of the pairwise relationship in joint embedding SSL.

4.2.1 Lung Sliding Classification

Multiple studies have explored the use of CNNs for automatically identifying absent lung sliding in LU M-mode images. The absence of lung sliding is a critical observation on LU because it indicates the possible presence of PTX, which is a life-threatening condition that worsens if unaddressed [93]. Although lung sliding can be observed on B-mode videos, it is subtle and the dynamic features are confined to the pleural line, which occupies a small fraction of B-mode LU images. Rather than use B-mode videos as input, studies seeking to automate lung sliding classification have leveraged M-mode images, as the distinction can be made on M-mode and the input space is reduced from videos to images. M-mode images can be extracted from B-mode videos by identifying a scan line and stacking the set of pixels within that line from each frame along a new axis.

Jaščur *et al.* [78] were the first to attempt the task of lung sliding classification with machine learning. Using a dataset of 48 videos acquired from 48 post-thoracic surgery patients with a single ultrasound probe, their model achieved a sensitivity of 0.82 and specificity of 0.92. VanBerlo *et al.* [158] developed a binary classifier using a dataset of 2535 videos acquired from hundreds of patients with variable probes. When evaluated on a test set of 540 examples, their model attained a sensitivity and specificity of 0.935 and 0.873, respectively. In a follow-up prospective validation study, they observed that the model retained its performance when applied to M-mode images collected directly from an ultrasound probe in a thoracic surgery ward (i.e., no M-mode extraction from B-mode) [47]. Lastly, Kim *et al.* [85] evaluated a workflow for assessing for lung sliding in B-mode videos that included modules for examination quality and identifying pleural line regions of interest. Similar to prior studies, they extracted M-mode images from B-mode videos for performing lung sliding classification.

4.2.2 Pairwise Relationship in Self-Supervised Learning

In the context of joint embedding methods, the *pairwise relationship* enforces constraints between examples that qualify them as a positive pair. Here we discuss the importance of the pairwise relationship. For a broader background on SSL, please refer to Section 2.3.2. Typically, a positive pair consists of two perturbations of a single example, where each perturbation is sampled from a distribution over image transformations. In their paper demonstrating how SimCLR improves performance and label efficiency in chest X-ray and dermatological image classification, Azizi *et al.* [9] suggested an alternative, situation-specific pairwise relationship – they considered any two distinct images of the same pathology to be a positive pair. For example, posteroanterior and lateral chest X-rays from the same patient encounter were a positive pair. A theoretical analysis by Balestriero and LeCun lent credence to the idea of context-specific positive pairs, providing justification that pretraining using SimCLR [29], Barlow Twins [180], or VICReg [12] should lead to similar or better performance on a downstream supervised learning task, as long as the pairwise relationship aligns with the labels for that task [11]. Their results underline the importance of medical knowledge in designing pretext tasks, motivating the definition of the M-mode pairwise relationship presented in this chapter.

4.3 Methods

4.3.1 Dataset

The datasets used for all training experiments in this chapter originated from a large, private LU database collected from two hospitals within an academic healthcare institution. Permission for data access is detailed in Appendix A. A portion of this database was previously labelled for the presence or absence of lung sliding by clinicians who had received training in LU as part of previous work [158]. All LU videos were divided into 3-second segments, with excess frames discarded. The resulting dataset, hereafter referred to as $\mathcal{D}_{\text{lab}}$, contained 4793 videos. $\mathcal{D}_{\text{lab}}$ was then randomly split by patient into three partitions: a training set of 3254 videos ($\mathcal{D}_{\text{train}}$), validation set of 743 videos ($\mathcal{D}_{\text{val}}$), and test set of 796 videos ($\mathcal{D}_{\text{test}}$). The database was queried for additional videos containing the A-line artefact but that were missing labels for lung sliding. This additional tranche of 14 249 unlabelled videos is referred to as $\mathcal{D}_{\text{unl}}$. Figure 4.2 illustrates the dataset split.

To investigate model generalizability, we evaluate on three additional datasets from external healthcare institutions with lung sliding class annotations: $\mathcal{D}_{\text{ext1}}$, $\mathcal{D}_{\text{ext2}}$, and $\mathcal{D}_{\text{ext3}}$.

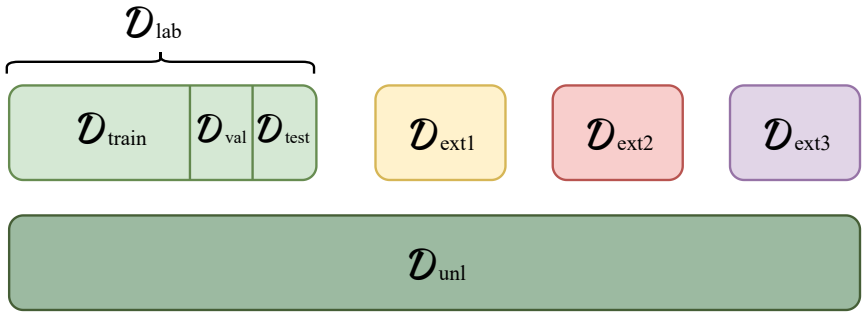


Figure 4.2: The datasets used in this chapter. $\mathcal{D}_{\text{lab}}$ is labelled for the downstream lung sliding classification task. It is split randomly by 70%/15%/15% into $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{val}}$, and $\mathcal{D}_{\text{test}}$, such that patients do not appear in multiple partitions.

		$\mathcal{D}_{\text{train}}$	$\mathcal{D}_{\text{val}}$	$\mathcal{D}_{\text{test}}$	$\mathcal{D}_{\text{unl}}$	$\mathcal{D}_{\text{ext1}}$	$\mathcal{D}_{\text{ext2}}$	$\mathcal{D}_{\text{ext3}}$
Lung sliding Present	Videos	2509	564	607	-	84	121	424
	Patients	366	76	72	-	25	55	155
Lung sliding Absent	Videos	745	179	189	-	26	30	77
	Patients	199	49	48	-	4	11	40
Total	Videos	3254	743	796	14 249	88	151	501
	Patients	565	125	120	3113	29	66	195

Table 4.1: Class decomposition for the datasets in this chapter, by ultrasound video.

Table 4.1 provides details on the label and patient decomposition of all labelled datasets.

M-mode Images from B-mode Videos

We followed a similar method outlined in [158] to extract synthetic M-mode images from retrospectively acquired greyscale B-mode videos. In clinical settings, M-mode images are acquired directly after designating a scan line while viewing a B-mode video stream. For the binary classification task, the vertical slice of the B-mode video with the maximum total pixel intensity was selected from all possible vertical slices within the horizontal bounds of the pleural line. This heuristic is consistent with the clinical notion that, in the majority of cases, the pleural line is the brightest object in a LU image of the upper and middle lobes of the lung. All M-mode images were resized to 128×128 pixels prior to pretraining. When pretraining, we retained the top 50% of each video's M-mode images, ordered by

total pixel intensity. As will be discussed in Section 4.3.2, the pairwise relationship for M-mode images required distinct M-mode images from the same video.

4.3.2 Self-Supervised Learning

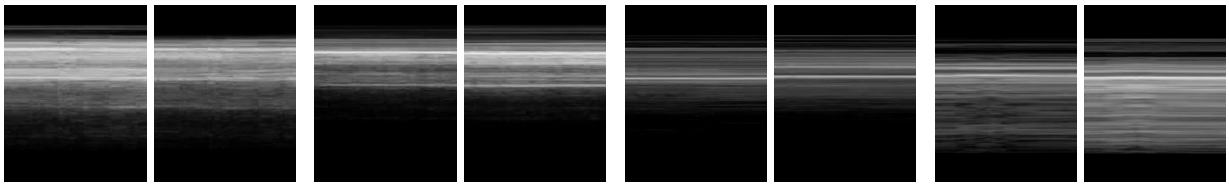
Pretext Tasks

We investigated three commonly employed joint embedding self-supervised pretraining techniques from the literature. SimCLR [29] is a contrastive learning method that employs a temperature-scaled cross entropy objective. We set the temperature to $\tau = 0.1$. Barlow Twins [180] is a non-contrastive learning method that maximizes distance between embeddings of pairs and includes an embedding decorrelation regularizer. We employ Barlow Twins with $\lambda = 0.005$ for the weight of the decorrelation regularizer. Inspired in part by Barlow Twins, VICReg [12] is a non-contrastive method that includes a regularizer that promotes high batch-wise variance for each dimension of the embeddings. We conducted pretraining trials with each of these methods. We set the weights of VICReg's three objective components to $\lambda = 25$, $\mu = 25$, and $\nu = 1$.

We pretrained for 100 epochs and used a batch size of 128 for all experiments. As in [9], we also investigated the effect of initializing the feature extractors with ImageNet-pretrained weights prior to self-supervised pretraining. After each pretraining experiment, weights of the projector were discarded, and the feature extractor was preserved for initialization in the downstream task. Consistent with prior work [158], we adopted the EfficientNetB0 [147] architecture as the feature extractor for all experiments, discarding the final block to reduce model capacity. In each experiment, the projector was a multilayer perceptron (MLP) with 3 layers of 128 nodes, with the rectified linear unit activation applied to each hidden layer.

Pairwise Relationship

As outlined in 4.3.1, the brightest 50% of the M-mode images within the bounds of the pleural line were utilized for pretraining. We considered any such M-mode images from the same video to be a positive pair. Qualitatively, different M-mode images produced from the same B-mode video appear similar. Crucially, they would have the same lung sliding label in the vast majority of cases, fulfilling the alignment condition outlined in [11]. Figure 4.3 displays examples of positive pairs of M-mode images. We fixed the pairwise relationship to focus on evaluating data augmentation transformations and the effects of



(a) Lung sliding, seashore sign. (b) Lung sliding, seashore sign. (c) Absent lung sliding, barcode sign. (d) Absent lung sliding, barcode sign

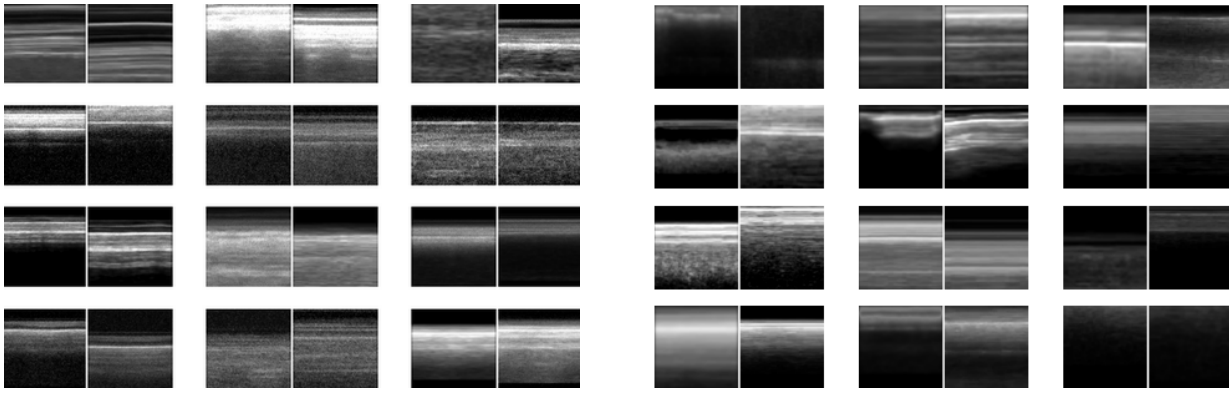
Figure 4.3: Examples of M-mode images satisfying the pairwise relationship of belonging to the same original B-mode video and intersecting the pleural line.

pretraining under different settings, relegating an ablation study for the M-mode pairwise relationship to future work.

Data Augmentation Strategy

A pretraining data augmentation strategy was devised to improve the invariance against inconsequential transformations and noise inherent to M-mode images. Transformations were identified that simulate natural variations present in M-mode LU but that do not impact the patterns exhibited by absent or present lung sliding, such as speckle noise and variable depth, gain, and contrast. The following series of stochastic transformations is applied to each M-mode image prior to subjecting it to the forward pass in pretraining:

1. With probability 0.8, we random crop of $c \sim \mathcal{U}(0.08, 1.0)$ of the image's area, which is resized to its original dimensions. To encourage the retention of the pleural line in the crop, the top of the crop was confined to the upper half of the image.
2. With probability 0.5, horizontal flip
3. With probability 0.5, Gaussian blur with a horizontal kernel of 10 pixels and $\sigma \sim \mathcal{U}(0.1, 2.0)$
4. With probability 0.5, random additive Gaussian noise is added to each pixel, sampled from $\mathcal{N}(\mu, \sigma^2)$, where $\mu \sim \mathcal{U}(-10, 10)$ and $\sigma \sim \mathcal{U}(0, 25)$
5. With probability 0.5, application of speckle noise, simulated using multiplicative Gaussian noise, sampled from $\mathcal{N}(1, \sigma^2)$, with $\sigma \sim \mathcal{U}(0, 0.1)$
6. With probability 0.8, brightness adjustment by $c \sim \mathcal{U}(-0.4, 0.4)$



(a) Custom M-mode augmentation pipeline (b) BYOL augmentation pipeline [59]

Figure 4.4: A selection of M-mode positive pairs that have been subjected to the M-mode-specific data augmentation pipeline.

7. With probability 0.8, contrast adjustment by $c \sim \mathcal{U}(-0.4, 0.4)$. With probability 0.5, contrast adjustment occurs before brightness adjustment.

In summary, a positive pair consisted of two M-mode images taken from a 3 s segment of the same original B-mode LU video that were each transformed via data augmentation. See Figure 4.4 for some examples of positive pairs.

4.3.3 Evaluation Protocol

This section describes the experimental protocols followed in this chapter. All experiments were conducted using a system with an Intel i9-10900K CPU at 3.7 GHz and a Nvidia GeForce RTX 3090 GPU.

Lung Sliding Classification

The downstream supervised learning task was the identification of absent lung sliding in M-mode images, which is a binary classification task. M-mode images of the upper and middle lobes of the lung were well suited for this problem, as there are established visual patterns employed by clinicians to distinguish between present and absent lung sliding. The seashore sign is indicative of lung sliding [92, 97], whereas the barcode/stratosphere sign signals the absence of lung sliding [92, 97]. We considered absent lung sliding to be the positive class.

Data Augmentation Pipeline Evaluation

We first ascertained the merit of the custom M-mode LU data augmentation pipeline for the pretext task (described in Section 4.3.2), as opposed to the ubiquitously cited augmentation pipeline for the Bootstrap Your Own Latent (BYOL) SSL pretraining method proposed by Grill *et al.* [59], which we accordingly refer to as the *BYOL augmentation pipeline*. Unlike the original BYOL study, we did not apply hue changes because LU M-mode images are greyscale. For each of the three studied pretraining methods, two CNN feature extractors were trained using $\mathcal{D}_{\text{train}}$ on the pretext task using the custom and BYOL augmentation pipelines respectively, producing a total of six feature extractors. A linear classifier was then appended and trained on the downstream task for each of the feature extractors. The best-performing augmentation pipeline across such experiments was then used for all subsequent pretraining trials.

Linear and Fine-tuning Evaluation

To investigate the effect of self-supervised pretraining on the performance in the downstream task, we adopted an evaluation protocol similar to those conducted by contemporary SSL publications that evaluate on natural images. We appended a single-node fully connected layer to the feature extractor with sigmoid activation.

We considered two methods to evaluate the utility of weights pretrained with SSL – linear modelling and fine-tuning. In the linear evaluation, the weights of the pretrained model were fixed and a linear classifier was trained using the feature representations from the extractor. In the case of fine-tuning, all model weights past the first block of the network were subject to updates. In both cases, we trained for 40 epochs with a learning rate of 1×10^{-4}, which was multiplied by $e^{-0.03}$ every epoch after the 15^{th} epoch. There was a heavy class imbalance in $\mathcal{D}_{\text{lab}}$ that favoured present lung sliding. As a result, we oversampled the minority class, taking the second, third, and fourth brightest M-mode images from each absent lung sliding video. Models were trained using the binary cross entropy loss function. Data augmentation was applied to training images, with random contrast reduction by $c \sim \mathcal{U}(0, 0.3)$, random brightness adjustment by $b \sim \mathcal{U}(-0.1, 0.1)$, additive Gaussian noise sampled from $\mathcal{N}(0, 5)$, and random horizontal flip. The model weights resulting in the lowest loss on $\mathcal{D}_{\text{val}}$ were saved for evaluation.

The performance of self-supervised pretrained models were compared against two fully supervised baselines, with weights initialized with either ImageNet-pretrained weights or with random values.

Label Efficiency

To assess the effects of SSL pretraining in its entirety, it was essential to consider its potency with respect to both performance differences and the ability to leverage otherwise unusable unlabelled datasets. Accordingly, we carried out a series of experiments that compared downstream performance under different levels of labelled data availability. We pretrained on $\mathcal{D}_{\text{train}}$ and fine-tuned using progressively larger subsets of $\mathcal{D}_{\text{train}}$. Fine-tuning was repeated using ImageNet-pretrained weights and randomly initialized weights, facilitating comparisons in low-label scenarios.

We conducted an additional experiment to explore the impact of pretraining with larger unlabelled datasets. Specifically, we pretrained on a large dataset consisting of $\mathcal{D}_{\text{lab}}$ and $\mathcal{D}_{\text{unl}}$ and evaluated performance on $\mathcal{D}_{\text{test}}$.

Explainability

Saliency maps (also referred to as "heatmaps") are a type of explanation for CNN predictions that consist of the original image superimposed onto a colour map indicating the regions that were most contributory to the prediction. We generated saliency maps using Grad-CAM [138] for selected predictions using models pretrained with SSL, initialized with ImageNet weights, and randomly initialized. A critical care physician possessing expertise in LU (hereafter referred to as the *LU expert*) compared the appropriateness of the saliency maps produced for pretrained and fully supervised models.

4.4 Results

4.4.1 Data Augmentation

We investigated the effect of applying our custom M-mode data augmentation pipeline in the pretraining phase. Six backbones were pretrained – for each of the three SSL methods examined, one backbone with randomly initialized weights and another with ImageNet-pretrained initial weights was pretrained. We executed linear evaluation trials and compared the AUC on $\mathcal{D}_{\text{val}}$, which is reported in Table 4.2. For backbones initialized with random weights, the custom M-mode augmentation pipeline resulted in higher performance in linear evaluation. Conversely, the BYOL augmentation pipeline exhibited a marked improvement compared to the M-mode-specific augmentations when backbones

	Random init.		ImageNet init.	
	BYOL	M-mode	BYOL	M-mode
SimCLR	0.585	0.658	0.864	0.827
Barlow Twins	0.554	0.578	0.826	0.818
VICReg	0.621	0.638	0.826	0.798

Table 4.2: AUC on $\mathcal{D}_{\text{val}}$ of linear classifiers trained using feature representations from self-supervised backbones pretrained using either the BYOL or the custom M-mode data augmentation pipeline.

were initialized using ImageNet-pretrained weights. As a result, all further experiments with backbones pretrained from scratch and those initialized with ImageNet-pretrained weights used the custom M-mode augmentations and BYOL augmentations, respectively.

4.4.2 Comparison with Supervised Baselines

To assess the quality of the pretrained representations, we conducted linear evaluation and fine-tuning trials using pretrained feature extractors. The results on $\mathcal{D}_{\text{test}}$ are compared with a linear classifier trained atop feature extractors initialized randomly and with ImageNet-pretrained weights. Tables 4.3 and 4.4 summarize the performance on $\mathcal{D}_{\text{test}}$ for backbones that were initialized with random weights and ImageNet-pretrained weights, respectively. Immediately apparent is the utility of initializing all feature extractors with ImageNet-pretrained weights, including those pretrained with SSL. Among the fine-tuning trials initialized with ImageNet weights, self-supervised pretrained models exhibited greater test performance. The results were less clear for linear evaluation using ImageNet weight initialization, as linear models using a frozen ImageNet-pretrained feature extractor achieved the greatest test AUC and sensitivity. We additionally found that in all cases where weights were initialized randomly, self-supervised pretrained models outperformed the fully supervised baselines.

	AUC		Specificity		Sensitivity		Accuracy	
Pretraining	LC	FT	LC	FT	LC	FT	LC	FT
SimCLR	0.645	**0.662**	0.582	0.755	0.619	0.476	0.590	0.688
Barlow Twins	**0.646**	0.634	0.644	0.718	0.534	0.460	0.618	0.657
VICReg	0.609	0.619	**0.926**	**0.860**	0.286	0.318	**0.774**	**0.731**
None	0.500	0.585	0.000	0.563	**1.000**	**0.540**	0.237	0.558

Table 4.3: Downstream task performance on $\mathcal{D}_{\text{test}}$, trained $\mathcal{D}_{\text{train}}$, reported for linear classifiers (LC) and fine-tuned models (FT). All backbones are initialized using random weights prior to pretraining. **Boldface** entries identify the pretraining method with the best performance.

	AUC		Specificity		Sensitivity		Accuracy	
Pretraining	LC	FT	LC	FT	LC	FT	LC	FT
SimCLR	0.742	0.861	**0.764**	**0.909**	0.497	0.545	**0.701**	**0.823**
Barlow Twins	0.707	**0.866**	0.705	0.845	0.598	**0.741**	0.680	0.820
VICReg	0.738	0.834	0.661	0.84	0.703	0.656	0.671	0.797
None	**0.768**	0.854	0.638	0.834	**0.751**	0.714	0.665	0.805

Table 4.4: Downstream task performance on $\mathcal{D}_{\text{test}}$, trained $\mathcal{D}_{\text{train}}$, reported for linear classifiers (LC) and fine-tuned models (FT). All backbones are initialized using ImageNet-pretrained weights prior to pretraining. **Boldface** entries identify the pretraining method with the best performance.

4.4.3 Label Efficiency

A major benefit of SSL is the ability to leverage unlabelled examples. To measure the effect of varying proportions of labelled data, we dropped fractions of the labels in $\mathcal{D}_{\text{train}}$ and conducted fine-tuning using networks pretrained with Barlow Twins on all examples in $\mathcal{D}_{\text{train}}$. Figure 4.5 details the results. The feature extractor trained from scratch benefited from self-supervised pretraining in the low-label setting. However, it appeared that this benefit was greatly diminished when the pretrained feature extractor and fully supervised feature extractor were both initialized with weights pretrained on ImageNet.

To further elucidate the benefit of pretraining with greater volumes of unlabelled data, we pretrained on both $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{unl}}$ using Barlow Twins and compared the performance to

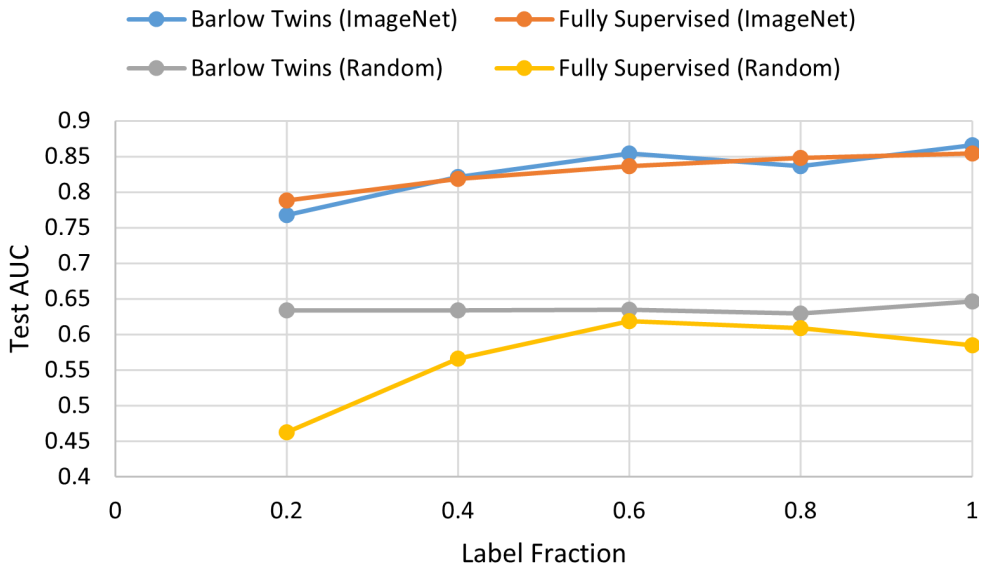


Figure 4.5: AUC on $\mathcal{D}_{\text{test}}$ for fine-tuned models initialized from weights pretrained on $\mathcal{D}_{\text{train}}$ using Barlow Twins and fully supervised models, initialized randomly or with ImageNet-pretrained weights.

pretraining with $\mathcal{D}_{\text{train}}$ alone. As shown in Table 4.5, more unlabelled data greatly improved the test performance of the feature extractors pretrained using random initialization, while performance dropped for those initialized with ImageNet weights.

Initialization	Pretraining data	$\mathcal{D}_{\text{test}}$ AUC
Random	$\mathcal{D}_{\text{train}}$	0.634
	$\mathcal{D}_{\text{train}} + \mathcal{D}_{\text{unl}}$	0.799
ImageNet	$\mathcal{D}_{\text{train}}$	0.866
	$\mathcal{D}_{\text{train}} + \mathcal{D}_{\text{unl}}$	0.838

Table 4.5: Downstream classification performance of models pretrained using Barlow Net using labelled and unlabelled datasets.

4.4.4 Evaluation on External Datasets

We evaluated all pretrained and fully supervised fine-tuned models initialized with ImageNet-pretrained weights on $\mathcal{D}_{\text{ext1}}$, $\mathcal{D}_{\text{ext2}}$, and $\mathcal{D}_{\text{ext3}}$. The results, presented in Table 4.6, did

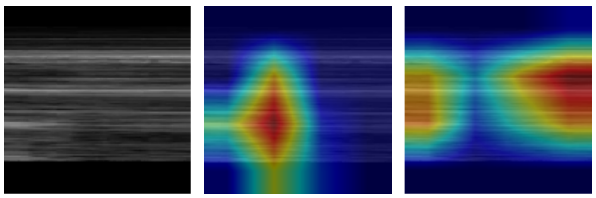
Pretrain Method	Pretraining Data	AUC	Specificity	Sensitivity	Accuracy
SimCLR	$\mathcal{D}_{\text{train}}$	0.765 [0.063]	0.765 [0.043]	0.532 [0.042]	0.729 [0.037]
	$\mathcal{D}_{\text{train}} + \mathcal{D}_{\text{unl}}$	0.778 [0.053]	**0.862** [**0.046**]	0.429 [0.103]	**0.834** [**0.056**]
Barlow Twins	$\mathcal{D}_{\text{train}}$	0.802 [0.071]	0.728 [0.043]	0.727 [0.138]	0.731 [0.048]
	$\mathcal{D}_{\text{train}} + \mathcal{D}_{\text{unl}}$	**0.833** [**0.059**]	0.761 [0.042]	0.723 [0.076]	0.756 [0.036]
VICReg	$\mathcal{D}_{\text{train}}$	0.807 [0.048]	0.727 [0.073]	**0.785** [**0.021**]	0.740 [0.059]
	$\mathcal{D}_{\text{train}} + \mathcal{D}_{\text{unl}}$	0.817 [0.053]	0.705 [0.076]	0.760 [0.114]	0.720 [0.054]
None	-	0.804 [0.076]	0.682 [0.097]	0.761 [0.075]	0.699 [0.071]

Table 4.6: Mean [std] performance across the three external datasets for pretrained and fully supervised models. Each backbone was initialized with ImageNet weights and pretrained using $\mathcal{D}_{\text{train}}$. Models pretrained using $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{unl}}$ outperformed those pretrained using $\mathcal{D}_{\text{train}}$ alone.

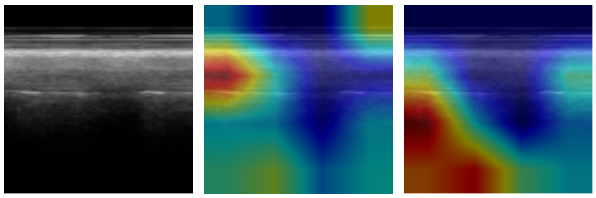
not indicate that the self-supervised models pretrained on $\mathcal{D}_{\text{train}}$ outperformed the fully supervised baseline when evaluated on datasets originating from other centres. However, with the exception of sensitivity, we found that performance on external datasets distinctly increased when models were pretrained on $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{unl}}$ combined, highlighting a potential benefit of leveraging unlabelled data when labels are available for a fraction of examples.

4.4.5 Explainability

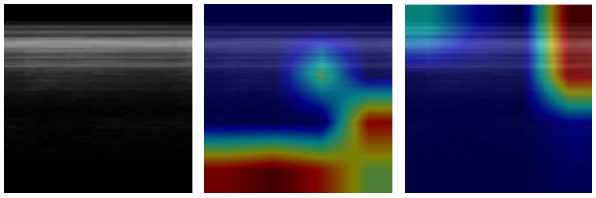
To explore the patterns learned by the fine-tuned models, we selected the pretrained network with the greatest $\mathcal{D}_{\text{test}}$ AUC (Barlow Twins) and produced saliency maps for 4 M-mode images in $\mathcal{D}_{\text{test}}$, using the pretrained network and a fully supervised network initialized with ImageNet-pretrained weights (see Figure 4.6). The LU expert reviewed the saliency maps without knowing which method was used to produce them and rated the appropriateness of the heatmaps using a 4-point Likert scale from 0 to 3, based on whether the highlighted regions corresponded to the areas of clinical interest. For instance, despite the false positive prediction by the self-supervised model shown in Figure 4.6a, the far field (bottom of the image) is dark, and the saliency map indicates activation by the bold straight lines in the near field, producing a prediction of absent lung sliding. The saliency maps generated for the pretrained model and fully supervised model scored averages of 2.5 and 1.5, respectively.



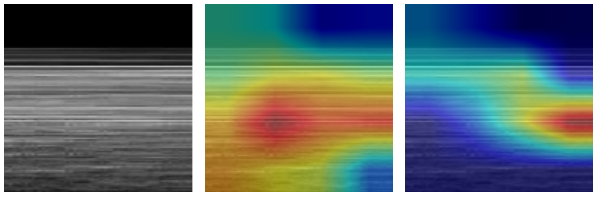
(a) False positive for both models.



(b) True negative for both models.



(c) False negative for the fully supervised model; true positive for the pretrained model.



(d) True positive for both models.

Figure 4.6: A selection of saliency maps for predictions produced by both a fully supervised model (left in each subfigure) and a model pretrained with Barlow Twins.

4.5 Discussion

The results of this chapter characterized the utility of self-supervised pretraining for M-mode LU examinations. First, it was clear that fine-tuning a self-supervised feature extractor provided improved initial feature representations compared to randomly initialized weights. The gap in performance on unseen examples strikingly narrowed when pretrained models and their fully supervised counterparts were initialized with the widely used ImageNet-pretrained weights. A similar trend was observed in Chapter 3 regarding the effectiveness of SSL pretraining for downstream tasks in B-mode LU. The results in this chapter indicated that, while there may be performance gains when fine-tuning SSL-pretrained models, the state-of-the-art contrastive and non-contrastive methods alone were not sufficient to reap substantial gains when trained on the labelled dataset and initialized with ImageNet-pretrained weights. However, when training from scratch, practitioners may benefit from pretraining.

A major insight derived from the results is that SSL-pretrained models improved performance on datasets from external centres for the task of lung sliding classification when using large volumes of unlabelled data that did not appear in the training set. This pattern was also observed in Chapter 3, potentially indicating a broader trend. Generalizability is a paramount concern for any organization deploying machine learning systems in health-

care settings. This finding is also consistent with the results of studies showing that SSL pretraining improves performance on external data in other medical imaging domains, such as chest X-ray classification [9,150], dermatologic image classification [9], and pathology slide classification [193]. Developers seeking to promote generalizability are therefore encouraged to employ self-supervised pretraining using any available unlabelled data.

Another noteworthy finding is that ImageNet-initialized feature extractors pretrained with the BYOL augmentation pipeline yielded better performance for the downstream classification task than ImageNet-initialized feature extractors pretrained using the custom M-mode augmentation pipeline. Again, the duality between ImageNet-pretrained and randomly initialized feature extractors manifested itself, as the M-mode augmentation pipeline produced better feature representations when pretrained models were initialized randomly. We conjectured that the BYOL augmentation pipeline's efficacy was related to the fact that the pretrained models were initialized with ImageNet-pretrained weights, since the original BYOL paper (and subsequent SSL publications, such as [12,180]) featured downstream classification evaluation on ImageNet and similar photographic datasets [59].

The experiments in this chapter had notable limitations. The training set consisted of LU examinations collected within a single healthcare institution, thereby limiting heterogeneity with respect to sources of variance such as device manufacturer, practitioner skill sets, and patient populations. Secondly, SimCLR performs best when larger batch sizes are employed during pretraining [29]; however, due to material limitations, we utilized a considerably small batch size. In keeping with the authors' findings, we trained for a large number of epochs to mitigate the impact of a small batch size. Lastly, the experiments did not produce classifiers that met or exceeded the performance metrics reported in previous work on automatic lung sliding classification [158]. However, unlike [158], we employed different datasets, we used standard binary cross entropy loss, we did not apply any techniques to mitigate overfitting, and we did not add fully connected layers between the feature extractor and the output layer. Rather than aiming to maximize the performance of the classifier, our focus was to study the effect of different SSL strategies and data augmentation distributions on the quality of representations.

In summary, this chapter constituted the first use of SSL in M-mode ultrasound. A selection of contemporary contrastive and non-contrastive SSL pretraining methods were investigated for the lung sliding classification task, using a pairwise relationship specific to M-mode LU. When evaluated on the downstream binary classification task of absent lung sliding detection, fine-tuned feature extractors initialized with self-supervised pretrained weights generally exhibited greater performance than fully supervised counterparts. Novel data augmentation specific to M-mode ultrasound were found to materially improve the performance of pretrained backbones whose weights were randomly initialized. Finally,

pretraining with larger unlabelled datasets resulted in improved metrics on evaluation datasets from external institutions, supporting the generalizability benefits conferred by SSL.

Although this chapter proposed a positive pair sampling strategy specific to LU M-mode, the M-mode images in each pair were selected at random from the set of images eligible for lung sliding assessment. In the next chapter, we introduce a clinically inspired, ultrasound-specific pairwise relationship for SSL that is applicable to both B-mode and M-mode ultrasound.

Chapter 5

Intra-video Positive Pairs in Self-Supervised Learning for Ultrasound

5.1 Introduction

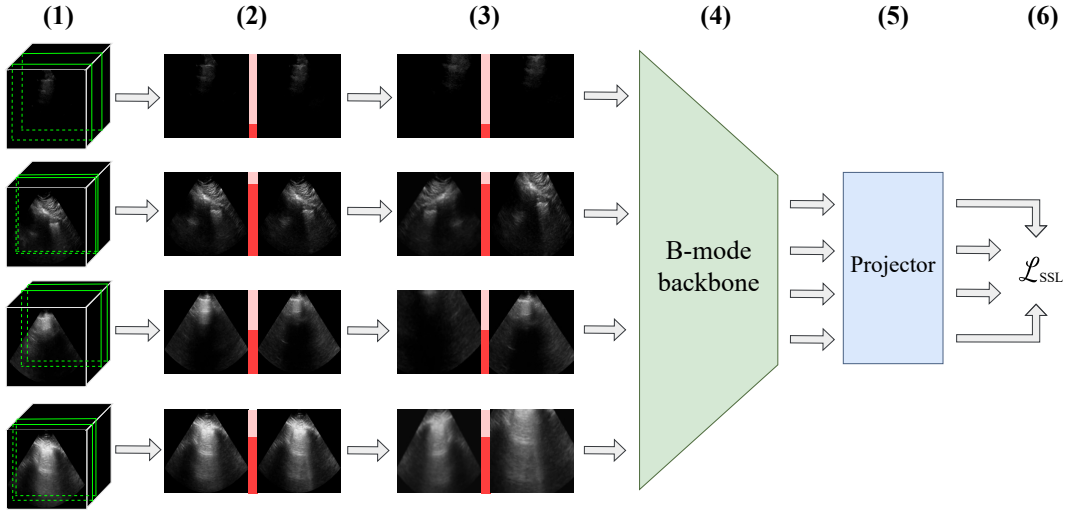
Lung ultrasound (LU) is a point-of-care diagnostic imaging technique that is safe, inexpensive, portable, and accurate for several respiratory conditions. Despite its benefits, there is a shortage of experts and clinical training programs [75, 136]. A multitude of machine learning solutions have been developed for lung ultrasound interpretation [6, 26, 42, 78, 85, 106, 158]. However, access to large, labelled datasets remains an obstacle. Point-of-care ultrasound (POCUS) examinations are not frequently documented in image archiving systems [63,84]. In cases where retrospective examinations are abundantly available, labelled datasets are still expensive to develop because interpretation expertise is scarce and expensive. In Chapters 3 and 4, we observed that self-supervised pretraining on unlabelled lung ultrasound (LU) images led to improved performance in core classification tasks, such as A-line versus B-line classification, pleural effusion detection, and lung sliding classification. Pretrained models also generally exhibited improved performance with smaller training sets and improved generalizability. However, the experiments in the prior chapters largely examined standard self-supervised learning (SSL) methods. In this chapter, we propose SSL methods tailored for ultrasound and assess them on LU classification tasks.

Unlike other forms of medical imaging, ultrasound is a dynamic modality acquired as a

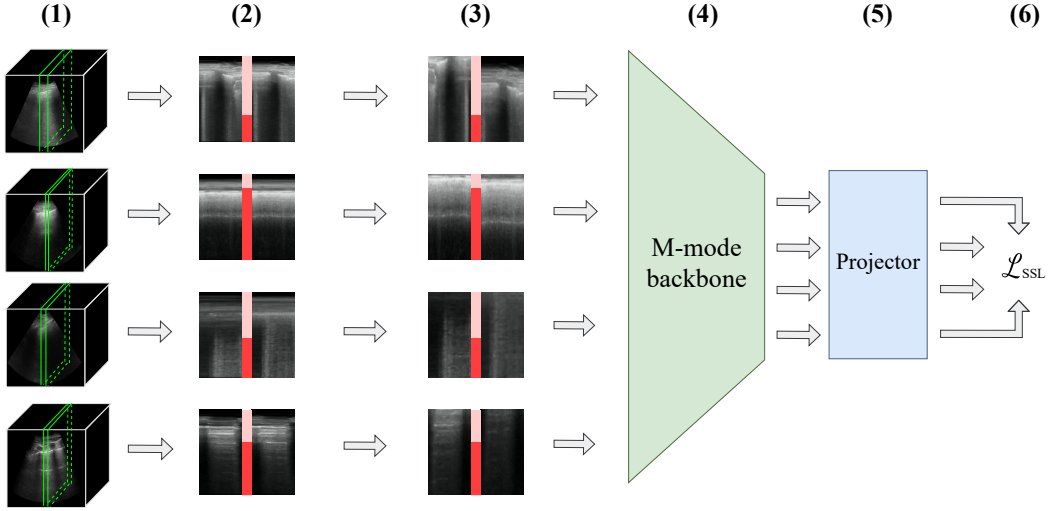
stream of frames, resulting in a video. Despite this, there are several ultrasound interpretation tasks that can be performed by assessing a still ultrasound image. Previous studies exploring SSL in ultrasound have exploited the temporal nature of ultrasound by defining contrastive learning tasks with intra-video positive pairs – positive pairs comprised of images derived from the same video [13, 33]. Recent theoretical results indicate that the pairwise relationship must align with the labels of the downstream task in order to guarantee that self-supervised pretraining leads to non-inferior performance on the downstream task [11]. For classification tasks, this means that positive pairs must have the same class label. Due to the dynamic nature of ultrasound, one cannot assume that all frames in an ultrasound video possess the same label for all downstream ultrasound interpretation tasks. As a result, it may be problematic to indiscriminately designate any pair of images originating from the same ultrasound video as a positive pair. Moreover, since ultrasound videos are often taken sequentially as a part of the same examination or from follow-up studies of the same patient, different ultrasound videos may bear a striking resemblance to each other. It follows that designating images from different ultrasound videos as negative pairs may result in negative pairs that closely resemble positive pairs.

In this chapter, we examined the effect of video proximity and sample weighting of intra-video positive pairs for common SSL methods. We also intended to determine if non-contrastive methods are more suitable for classification tasks in ultrasound. Since non-contrastive methods do not require the specification of negative pairs, we conjectured that non-contrastive methods would alleviate the issue of cross-video similarity and yield stronger representations for downstream tasks. Our contributions and results are summarized as follows:

- A method for sampling intra-video positive pairs for joint embedding SSL with ultrasound.
- A sample weighting scheme for joint embedding SSL methods that weighs positive pairs according to the temporal or spatial distance between them in their video of origin
- A comprehensive assessment of intra-video positive pairs integrated with SSL pretraining methods, as measured by downstream performance in B-mode and M-mode lung ultrasound classification tasks. We found that, with proper downstream task-specific hyperparameters, intra-video positive pairs improves performance compared to the standard practice of producing two distortions of the same image.
- A comparison of contrastive and non-contrastive learning for multiple lung ultrasound



(a) For B-mode ultrasound, positive pairs are temporally separated images from the same video.



(b) For M-mode ultrasound, positive pairs are spatially separated images from the same video.

Figure 5.1: An overview of the methods introduced in this chapter. Positive pairs of images separated by no more than a threshold are sampled from the same B-mode video **(1)**. Sample weights inversely proportional to the separation between each image (red bars) are calculated for each pair **(2)**. Random transformations are applied to each image **(3)**. Images are sent to a neural network consisting of a backbone **(4)** and a projector **(5)** connected in series. The outputs are used to calculate the self-supervised objective $\mathcal{L}_{\text{SSL}}$ **(6)**. The backbone is retained for downstream supervised learning tasks.

classification tasks. Contrary to our initial belief, a contrastive method outperformed multiple non-contrastive methods on multiple lung ultrasound downstream tasks.

Figure 5.1 encapsulates the novel methods proposed in this chapter. To our knowledge, there are no preceding studies that systematically investigate the effect of sampling multiple images from the same ultrasound video in non-contrastive learning. More generally, we believe that this chapter constitutes the first attempt to integrate sample weights into non-contrastive objectives.

5.2 Related Works

The methods in this chapter fall under the category of joint embedding SSL. For a comprehensive background on these methods, please refer to Section 2.3.

Ultrasound is a dynamic imaging modality that is typically captured as a sequence of images and stored as a video. As such, images originating from the same video are highly correlated and are likely to share semantic content. Accordingly, recent works have developed ultrasound-specific contrastive learning methods that construct positive pairs from the same video. While Section 2.4 describes the applications of SSL in LU, we elucidate some key methods designed for ultrasound. The Ultrasound Contrastive Learning (USCL) method [33] is a derivative of SimCLR in which positive pairs are weighted sums of random images within the same video (i.e., the mixup operation [184]), while negative pairs are images from different videos. They reported an improvement on COVID-19 classification with the POCOVID-Net dataset [19]. Although it was originally proposed as a semi-supervised learning method, an ablation study that used only the contrastive learning loss during pretraining demonstrated improved performance on POCOVID-Net. Improving on USCL, Meta-USCL concurrently trains a separate network that learns to weigh positive pairs [32]. The work was inspired by the observation that the intra-video positive pairs may exhibit a wide range of semantic similarity or dissimilarity. Basu *et al.* proposed a MoCo-inspired solution where positive pairs are images that are temporally close within a video, while negative pairs consist of either pairs from different videos or pairs from the same video that are separated temporally by no less than a threshold that is gradually decreased over the course of training [13]. Lastly, the HiCo method's objective is the sum of a softened InfoNCE loss calculated for the feature maps outputted by various model blocks [182] The authors reported greatly improved performance with respect to USCL.

Standard non-contrastive methods have been applied for various tasks in ultrasound imaging. In addition to assessing contrastive methods, Anand *et al.* conducted pretraining with two self-distillation non-contrastive methods (BYOL [59] and DINO [24]) on a large dataset of echocardiograms [4]. BYOL pretraining has also been applied in anatomical tracking tasks [90]. Information maximization methods have been investigated for classification tasks in M-mode and B-mode lung ultrasound [155,156]. To our best knowledge, no studies have trialled non-contrastive learning methods for B-mode ultrasound with intra-video positive pairs. The methods introduced in the current chapter sought to address this gap in the literature by investigating the effect of sampling positive pairs from the same video on the efficacy of non-contrastive pretraining for tasks in ultrasound.

5.3 Methods

5.3.1 Joint Embedding Methods with Intra-Video Positive Pairs

Setup

Consider the standard joint embedding scenario where unlabelled data are provided and the goal is to maximize the similarity between embeddings of positive pairs. In contrastive learning, maximizing the dissimilarity of negative pairs is a component of the objective. Let $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ denote a positive pair of ultrasound images. Self-supervised pretraining results in a feature extractor $f(\boldsymbol{x})$ that outputs representation vector $\boldsymbol{h}$. The goal of SSL is to produce a backbone whose weights constitute a better starting point for learning the downstream task than random weight initialization.

In this chapter, we propose a method for sampling and weighing positive pairs in the joint embedding setting that can be adopted for any joint embedding SSL method. Hereafter, we refer to the proposed method as IVPP. We adopted SimCLR [29], Barlow Twins [180] and VICReg [12] for our experiments. In these methods, an MLP projector is appended to the feature extractor during pretraining. $\boldsymbol{z} = g(\boldsymbol{h}) = g(f(\boldsymbol{x}))$ is the embedding vector outputted by the projector. The SSL objective is then computed in embedding space.

Intra-video Positive Pairs

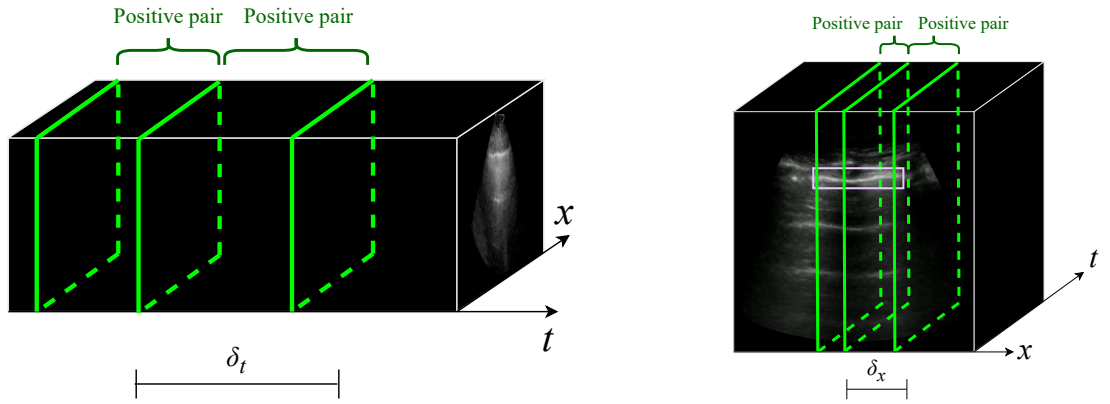
Recall that positive pairs are images that are semantically related. Previous work in contrastive SSL for ultrasound has explored the use of intra-video positive pairs [13, 32,

33, 182]. A problem with naively sampling intra-video positive pairs is that it rests on the assumption that all images in the video are sufficiently similar. However, clinically relevant signs commonly surface and disappear within the same ultrasound video as the probe and/or the patient move. For example, B-lines are artefacts in lung ultrasound that signify diseased lung parenchyma [143]. B-lines may disappear and reappear as the patient breathes or as the sonographer moves the probe. The A-line artefact appears in the absence of B-lines, indicating normal lung parenchyma. In the absence of patient context, an image containing A-lines and an image containing B-lines from the same video convey quite different impressions. While most previous methods only considered inter-video images to be negative pairs, Basu *et al.* [13] argued that that temporally distant intra-video pairs of ultrasound images are more likely to be dissimilar, which inspired their method that treats such instances as negative pairs. Despite this, we argue that distant intra-video images may sometimes exhibit similar content, and therefore should not be treated as negative pairs by default. For example, the patient and probe may remain stationary throughout the video, or the probe may return to its original position and/or orientation. Moreover, periodic physiological processes such as the respiratory cycle may result in temporally distant yet semantically similar images. Without further knowledge of the ultrasound examinations in a dataset, we conjectured that it may be safest to only assume that positive pairs are intra-video images that are close to each other. Closer pairs are likely to contain similar semantic content, yet they harbour different instantiations of noise that models should be invariant to. In summary, this method distinguishes itself from prior work by only considering proximal frames to be positive pairs and treating distant intra-video pairs as neither positive nor negative pairs.

For B-mode videos, we defined positive pairs as intra-video images $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ that are temporally separated by no more than δ_t seconds. To accomplish this, $\boldsymbol{x}_1$ is randomly drawn from the video's images, and $\boldsymbol{x}_2$ is randomly drawn from the set of images that are within δ_t seconds of $\boldsymbol{x}_1$. The frame rate of the videos must be known in order to determine which images are sufficiently close to $\boldsymbol{x}_1$. Note that videos with higher frame rates provide more candidates for positive pairs, potentially increasing the diversity of pairs with respect to inherently occurring noise. Figure 5.2 conveys how positive pairs are sampled.

A similar sampling scheme was developed for M-mode images. Like previous studies, we extracted synthetic M-mode images as vertical slices through time of a retrospectively acquired B-mode video, taken at a specific x-coordinate in the video [78, 156, 158]. The x-axis of an M-mode image is time, and its y-axis is the vertical dimension of the B-mode video. We define positive pairs to be M-mode images whose x-coordinates differ by no more than δ_x pixels. To avoid resolution differences, all B-mode videos are resized to the same width and height prior to sampling M-mode images.

As is customary in joint embedding methods, stochastic data augmentation is applied to each image, encouraging the feature extractor to become invariant to semantically insignificant differences. Any data augmentation pipeline may be adopted for this formulation of intra-video positive pairs; however, we recommend careful selection of transformations and the distributions of their parameters to ensure that the pairwise relationship continues to be consistent with the downstream ultrasound task.



(a) For B-mode ultrasound, positive pairs are frames in the same video that are within δ_t seconds of each other.

(b) For M-mode ultrasound, positive pairs are synthetic M-mode images produced by extracting vertical slices of pixels from the same B-video that are horizontally separated by no more than δ_x pixels. In the context of lung ultrasound, M-mode images should intersect the pleural line (outlined in mauve).

Figure 5.2: Illustration of intra-video positive pairs. Positive pairs are considered images that are no more than a threshold apart from each other within the same ultrasound video.

Sample Weights

The chance that intra-video images are semantically related decreases as temporal or spatial separation increases. To temper the effect of unrelated positive pairs, we applied sample weights to positive pairs in the SSL objective according to their temporal or spatial distance. Distant pairs were weighed less than closer pairs. For a positive pair of B-mode images occurring at times t_1 and t_2, or for M-mode images occurring at horizontal positions x_1 and x_2, sample weight is calculated as follows:

$$w = \frac{\delta_t - |t_2 - t_1| + 1}{\delta_t + 1} \qquad w = \frac{\delta_x - |x_2 - x_1| + 1}{\delta_x + 1} \tag{5.1}$$

Sample weights were incorporated into each SSL objective trialled in this chapter. Accordingly, we modified the objective functions for SimCLR, Barlow Twins, and VICReg in order to weigh the contribution to the loss differently based on sample weights. Appendix C describes the revised objective functions. To our knowledge, this is the first time that sample weighting schemes have been proposed for the aforementioned self-supervised learning methods.

5.3.2 Ultrasound Classification Tasks:

5.3.3 COVID-19 Classification (COVID)

As was done in previous studies on ultrasound-specific joint embedding methods [13, 32, 33, 182], we evaluate IVPP on the public POCOVID-Net lung ultrasound dataset [19]. This dataset contains 140 publicly sourced videos (2116 images) labelled for three classes: COVID-19 pneumonia, non-COVID-19 pneumonia, and normal lung.[1] When evaluating on POCOVID-Net, we pretrain on the public Butterfly dataset, which contains 22 unlabelled lung ultrasound videos [22].[2]

A-line Versus B-line Classification (AB)

A-lines and B-lines are two cardinal artefacts in B-mode LU that can provide quick information on the status of a patient's lung tissue. A-lines are reverberation artefacts that are indicative of normal, clear lung parenchyma [143]. On LU, they appear as horizontal lines deep to the pleural line. Conversely, B-lines are indicative of diseased lung tissue [143]. Generally, the two are mutually exclusive. We evaluated on the binary classification task of A-lines versus B-lines on LU, as was done in previous work benchmarking joint embedding SSL methods for LU tasks [155].

[1] See dataset details at the public repository [19]:
https://github.com/jannisborn/covid19_ultrasound.

[2] Accessed via a URL available at the public USCL repository [33]:
https://github.com/983632847/USCL.

		Unlabelled	Labelled		
			Train	Validation	Test
Total	Patients	5204	1540	330	329
	Videos	20 000	4123	858	936
	B-Mode Images	4 611 063	927 889	191 437	208 648
AB Labels	Videos	–	2100 / 998	441 / 197	512 / 213
	B-Mode Images	–	484 287 / 216 505	99 132 / 40 608	116 648 / 42 122
LS Labels	Videos	–	3135 / 465	605 /115	683 / 129
	M-Mode Images	–	331 246 / 49 053	64 064 / 10 978	70 835 / 11 236

Table 5.1: Breakdown of ParenchymalLUS at the video and image level. The number of labelled videos and constituent images are indicated for the A-lines versus B-lines task (AB) and the lung sliding classification task (LS). x / y indicates the number of negative and positive labelled examples available for each task, respectively. Video labels apply to each image within the video. Note that some videos were not labelled for both tasks.

We used a private dataset of 25 917 parenchymal LU videos (5.9×10^6 images), hereafter referred to as *ParenchymalLUS*. It was a subset of a larger database of de-identified LU videos that was partially labelled for previous work [6, 158]. Access to this database is detailed in Appendix A. Before experimentation, we split the labelled portion of ParenchymalLUS by anonymous patient identifier into training, validation, and test sets. The unlabelled portion of ParenchymalLUS was assembled by gathering 20 000 videos from the unlabelled pool of videos in the database that were predicted to contain a parenchymal view of the lungs by a previously trained LU view classifier [152]. All videos from the same patient were in either the labelled or the unlabelled subset. Table 5.1 provides further information on the membership of ParenchymalLUS.

Lung Sliding Classification (LS)

Lung sliding is a dynamic sign that, when observed on a parenchymal LU view, rules out the possibility of a pneumothorax at the site of the probe [95]. The absence of lung sliding is suggestive of pneumothorax, warranting further investigation. On B-mode ultrasound, lung sliding manifests as a shimmering of the pleural line [95]. The presence or absence of lung sliding is also appreciable on M-mode LU images that intersect the pleural line [92, 97]. We evaluated on the binary lung sliding classification task, where positive pairs were M-mode images originating from the same B-mode video that intersected the pleural line.

ParenchymalLUS was adopted for the lung sliding classification task. We used the same train/validation/test partition as described above. Following prior studies, we estimated the horizontal bounds of the pleural line using a previously trained object detection model [158] and used the top half of qualifying M-mode images, in decreasing order of total pixel intensity [156].

5.4 Results

5.4.1 Training Protocols

Unless otherwise stated, all feature extractors were initialized with ImageNet-pretrained weights. Similar studies concentrating on medical imaging have observed that this practice improves downstream performance when compared to random initialization [9, 156]. Moreover, we designated fully supervised classifiers initialized with ImageNet-pretrained weights as a strong baseline against which to compare models pretrained with SSL.

Evaluation on POCOVID-Net followed a similar protocol employed in prior works [13, 33]. Feature extractors with the ResNet18 architecture [68] were pretrained on the Butterfly dataset. Prior to training on the POCOVID-Net dataset, a 3-node fully connected layer with softmax activation was appended to the pretrained feature extractor. Five-fold cross validation was conducted with POCOVID-Net by fine-tuning the final three layers of the pretrained feature extractor. Unlike prior works, we adopted the average across-folds validation accuracy, instead of taking the accuracy of the combined set of validation set predictions across folds. Presenting the results in this manner revealed the high variance of model performance across folds, which may be due to the benchmark dataset's small video sample size.

All experiments with ParenchymalLUS utilized the MobileNetV3-Small architecture as the feature extractor, which outputs a 576-dimensional representation vector [71]. Feature extractors were pretrained on the union of the unlabelled videos and labelled training set videos in ParenchymalLUS. Performance was assessed via test set classification metrics. Prior to training on the downstream task, a single-node fully connected layer with sigmoid activation was appended to the pretrained feature extractor. We report the performance of linear classifiers trained on the frozen feature extractor's representations, along with classifiers that are fine-tuned end-to-end.

For each joint embedding method, the projectors were MLPs with two 768-node layers, outputting 768-dimensional embeddings. Pretraining was conducted for 500 epochs using

the LARS optimizer [177] with a batch size of 384 and a learning rate schedule with warmup and cosine decay as in [12].

B-mode and M-mode images were resized to 224×224 and 224×112 pixels respectively using bilinear interpolation. The width of M-mode images was standardized to a smaller value was because the height of B-mode images often far exceeded the number of frames in a 3-second segment of B-mode video. Since feature extractors were initialized with ImageNet-pretrained weights, pixel intensities were mean-centered and normalized using the mean and standard deviation of the ImageNet dataset.

All IVPP pretraining runs applied stochastic data augmentation to intra-video positive pairs. Each image was subjected to the following sequence of stochastic transformations for data augmentation:

1. Randomly located crop of a fraction of the image's area in the range $[0.4, 1.0]$, followed by resizing to the original image dimensions. For B-mode images, the width/height aspect ratio was confined to the range $[0.8, 1.25]$, while M-mode crops were confined to $[0.4, 0.6]$.
2. With probability 0.5, horizontal reflection.
3. With probability 0.5, random brightness change in the range $[-0.25, 0.25]$.
4. With probability 0.5, random contrast change in the range $[-0.25, 0.25]$.
5. With probability 0.25, random Gaussian blur with a kernel size of 5 pixels and a standard deviation uniformly sampled from the range $[0.1, 2.0]$.

Training runs for the `COVID` and `AB` tasks with B-mode images also utilized this data augmentation pipeline. For `LS` training runs with M-modes, the minimum allowable crop area was increased to 95% of the image's area to ensure the pleural line was almost always visible.

All pretraining and training runs were conducted using virtual machines equipped with an Intel E5-2683 v4 Broadwell CPU at 2.1 GHz and a Nvidia Tesla P100 GPU with 12 GB of VRAM. Python 3.10 was utilized for all IVPP experiments. The official USCL[3] and UCL[4] source code repositories were adapted for executing experiments comparing them to IVPP. Source code for the experiments in this chapter is available in a public GitHub repository.[5]

[3] https://github.com/983632847/USCL
[4] https://github.com/gbc-iitd/US_UCL
[5] https://github.com/bvanberl/IVPP

5.4.2 Performance

The two main proposed features of IVPP are intra-video positive pairs and distance-based sample weights. Accordingly, we assessed the performance of IVPP across multiple assignments of the maximum image separation. Separate trials were conducted for SimCLR, Barlow Twins, and VICReg pretraining. For the `COVID` and `AB` tasks, we explored the values $\delta_t \in \{0, 0.5, 1, 1.5\}$ seconds. The `LS` task is defined for M-mode ultrasound, and so we explored $\delta_x \in \{0, 5, 10, 15\}$ pixels. The standardized width of B-mode videos should be considered when determining an appropriate range for δ_x. Note that when $\delta = 0$, sample weights are all 1, thereby keeping the SSL objectives unaltered.

Figure 5.3 summarizes the performance of IVPP on the public POCOVID-Net dataset after pretraining on the Butterfly dataset, which is measured by average test accuracy in 5-fold cross validation. In most cases, pretrained models obtained equal or greater average accuracy than the ImageNet-pretrained baseline, with the exception of Barlow Twins with $\delta_t = 0$ and $\delta_t = 0.5$. The performance of models pretrained with SimCLR, Barlow Twins, and VICReg peaked at different nonzero values of δ_t (0.5, 1, and 1.5 respectively), indicating a possible benefit of selecting temporally close yet distinct intra-video positive pairs. It was also observed across all three pretraining methods that the inclusion of sample weights resulted in worsened test AUC when $\delta = 0.5$, but improved test AUC when $\delta = 1.0$ and $\delta = 1.5$.

Similar experiments were conducted with ParenchymalLUS for the `AB` task and `LS` task, using B-mode and M-mode images respectively as input. ParenchymalLUS represents a scenario where there is an abundance of unlabelled data, which differs greatly from the preceding evaluation on small, public datasets. The unlabelled and labelled portions of ParenchymalLUS contained at least an order of magnitude more videos than either of the public Butterfly or POCOVID-Net datasets. B-mode and M-mode feature extractors were pretrained on the union of the unlabelled and training portions of ParenchymalLUS — one for each value of δ, with and without sample weights. For these evaluations, we used all training examples that were previously assigned a label for the downstream task. Figure 5.4 provides a visual comparison of the test AUC obtained by linear feature representation classifiers and fine-tuned models for the `AB` and `LS` tasks. Numeric results are available in Appendix D. One trend observed across both tasks and evaluation types was that SimCLR, a contrastive method, consistently outperformed Barlow Twins and VICReg, which are both non-contrastive methods. Furthermore, pretraining with non-contrastive methods often resulted in worse test AUC compared to initialization with ImageNet-pretrained weights. Another observation across all experiments was that there was no discernible trend for the effect of sample weights that was consistent for any task, pretraining method,

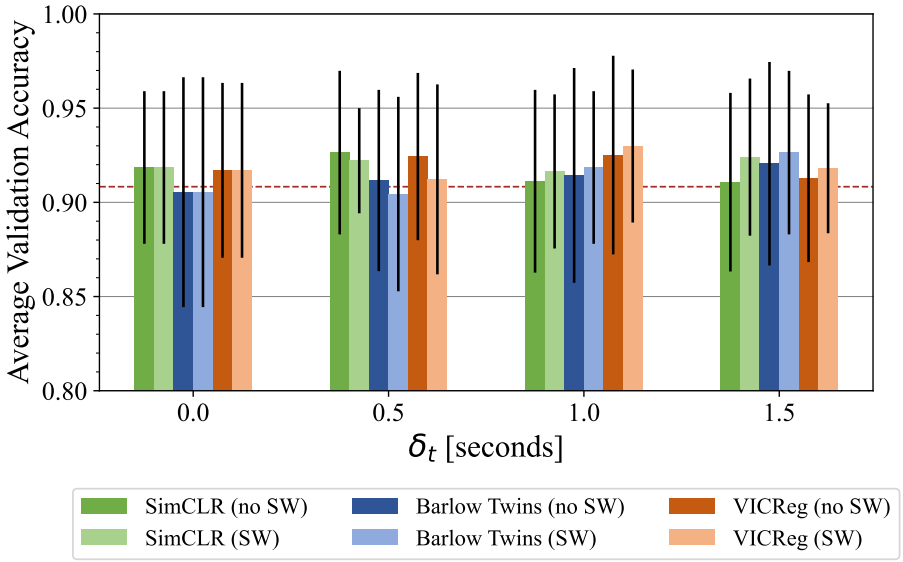


Figure 5.3: Average test accuracy across 5-fold cross validation on the POCOVID-Net dataset. Models were pretrained with a variety of intra-video positive pair thresholds with and without sample weights. Error bars indicate the standard deviation across folds. The dashed line indicates initialization with ImageNet-pretrained weights.

δ_t, or δ_x.

Focusing on AB, linear classifiers achieved the greatest performance when $\delta_t > 0$, with the exception of VICReg (Figure 5.4a). The use of SimCLR compared to the other pretraining methods appeared to be responsible for the greatest difference in test performance. As shown in Figure 5.4a, SimCLR-pretrained models outperformed non-contrastive methods, and were the only models that outperformed ImageNet-pretrained weights. The use of a nonzero δ_t resulted in slight improvement in combination with SimCLR pretraining, but degraded performance of non-contrastive methods.

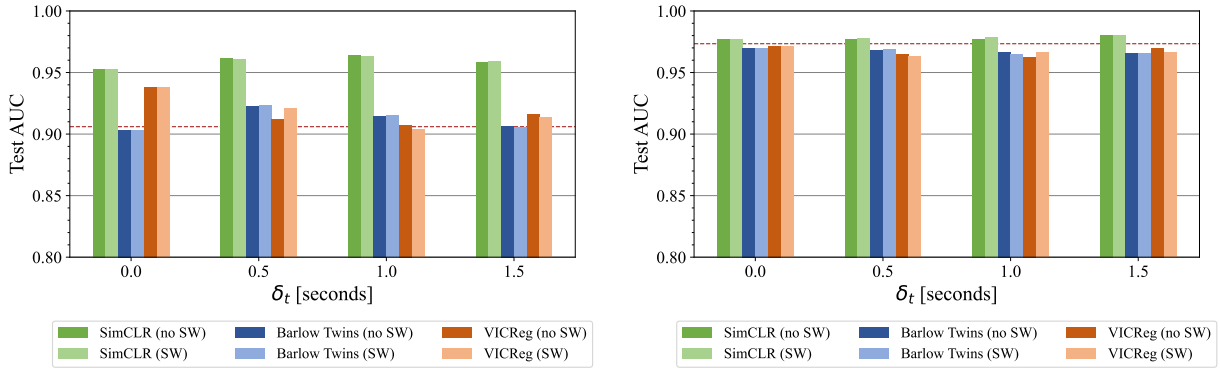
Similar results were observed for the LS M-mode classification task. Models pretrained with SimCLR were the only ones that matched or surpassed fully supervised models. Nonzero δ_x generally improved the performance of linear classifiers, with $\delta_x = 5$ pixels corresponding to the greatest test AUC for SimCLR and VICReg, and $\delta_x = 15$ for Barlow Twins. Inclusion of sample weights appreciably improved the performance of Barlow Twins-pretrained models. Fine-tuned models pretrained with SimCLR performed similarly to fully supervised models, while non-contrastive methods resulted in worsened test AUC.

Table 5.2 compares the top-performing IVPP-pretrained models for each SSL method with two prior ultrasound-specific contrastive learning methods — USCL [33] and UCL [13]. Notably, all three self-supervised methods pretrained with IVPP outperformed ImageNet-pretrained initialization for POCOVID-Net, a task where very little pretraining and training data were utilized. For the B-mode and M-mode tasks assessed with ParenchymalLUS, a contrastive method (including the baseline) outperformed non-contrastive methods. Appendix E provides additional results that showed a similar trend with different pretraining batch sizes. Overall, the most salient result from the above experiments was that SimCLR — a contrastive learning method — outperformed both non-contrastive methods when unlabelled data was abundant.

5.4.3 Label Efficiency

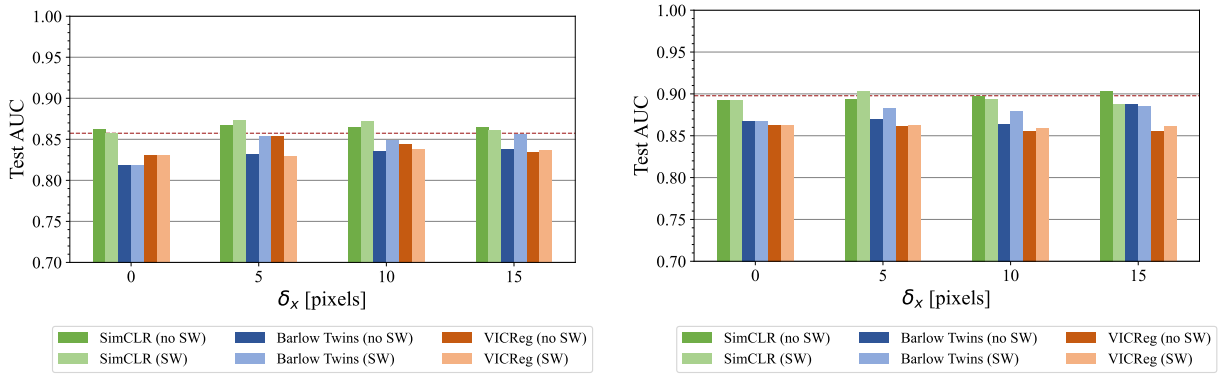
ParenchymalLUS is much larger than public ultrasound machine learning datasets. Although the majority of its videos are unlabelled, it contains a large number of labelled examples. To simulate a scenario where the fraction of examples that are labelled is much smaller, we investigated the downstream performance of models that were pretrained on all the unlabelled and training ParenchymalLUS examples and then fine-tuned on a small subset of the training set.

Label efficiency investigations are typically conducted by fitting a model for the downstream task using progressively smaller fractions of training data to gauge how well self-



(a) Linear classifiers for the `AB` task

(b) Fine-tuned classifiers for the `AB` task



(c) Linear classifiers for the `LS` task

(d) Fine-tuned classifiers for the `LS` task

Figure 5.4: ParenchymalLUS test set AUC for the `AB` and `LS` binary classification tasks, calculated for models pretrained with a selection of contrastive and non-contrastive learning methods and employing a variety of intra-video positive pair thresholds with and without sample weights (SW). The dashed line indicates initialization with ImageNet-pretrained weights.

Dataset	POCOVID-Net	ParenchymalLUS	
Pretraining method	Mean (std) test accuracy	A/B Test AUC	LS Test AUC
Random initialization	0.881 (0.050)	0.954	0.790
ImageNet initialization	0.908 (0.043)	0.973	0.898
USCL [33]	0.905 (0.044)	0.979	0.874
UCL [13]	0.901 (0.054)	0.967	0.809
IVPP [SimCLR]	0.926 (0.043)	0.980	0.903
IVPP [Barlow Twins]	0.921 (0.054)	0.969	0.887
IVPP [VICReg]	0.930 (0.046)	0.971	0.862

Table 5.2: Performance of fine-tuned models pretrained using IVPP compared to ultrasound-specific contrastive learning methods, USCL and UCL, and to baseline Random and ImageNet initializations.

supervised models fare in low-label scenarios. The results of these experiments may be unique to the particular training subset that is randomly selected. We designed an experiment to determine if the choice of δ_t, δ_x, or the introduction of sample weights influenced downstream performance in low-label settings. To reduce the chance of biased training subset sampling, we divided the training set into 20 subsets and repeatedly performed fine-tuning experiments on each subset for each pretraining method and δ value, with and without sample weights. Inspection of the central moments and boxplots from each distribution indicated that the normality and equal variance assumptions for analysis of variance (ANOVA) were not violated. For each pretraining method, a two-way repeated-measures ANOVA was performed to determine whether the mean test AUC scores across values of δ and sample weight usage were different. The independent variables were δ and the presence of sample weights, while the dependent variable was the AUC on the test set. Whenever the null hypothesis of the ANOVA was rejected, post-hoc paired t-tests were performed to compare the following:

- Pretraining with nonzero δ against standard positive pair selection ($\delta = 0$)
- For the same nonzero δ value, sample weights against no sample weights

For each group of post-hoc tests, the Bonferroni correction was applied to establish a family-wise error rate of $\alpha = 0.05$. To ensure that each training subset was independent, we split the dataset by anonymous patient identifier. This was a necessary step because intra-video images are highly correlated, along with videos from the same patient. As a

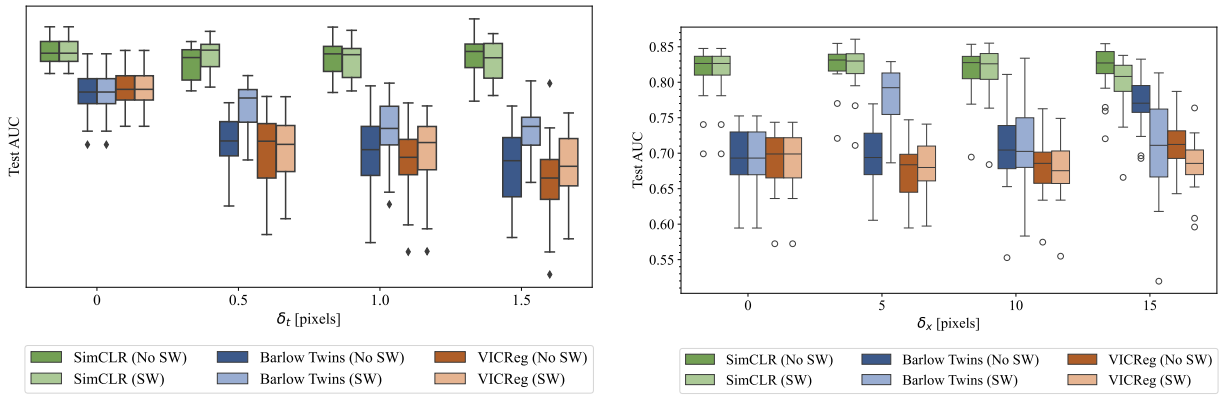
result, the task became substantially more difficult than naively sampling 5% of training images because the volume *and* heterogeneity of training examples was reduced by training on a small fraction of examples from a small set of patients.

The fine-tuning procedure was identical to that described in section 5.4.1, with the exception that the model's weights at the end of training were retained for evaluation, instead of restoring the best-performing weights on the validation set. Figure 5.5 provides boxplots for all trials that indicate the distributions of test AUC under the varying conditions for both the `AB` and `LS` tasks. Again, SimCLR performance appeared to be substantially higher than both non-contrastive methods.

Table 5.3 gives the mean and standard deviation of each set of trials, for each hyperparameter combination. For each task and each pretraining method, the ANOVA revealed significant interaction effects ($p \leq 0.05$). Accordingly, all intended post-hoc t-tests were performed to ascertain (1) which combinations of hyperparameters were different from the baseline setting of augmenting the same frame twice ($\delta = 0$) and (2) values of δ where the addition of sample weights changes the outcome. First, we note that SimCLR was the only pretraining method that consistently outperformed full supervision with ImageNet-pretrained weights. Barlow Twins and VICReg pretraining — both non-contrastive methods — resulted in worse performance.

For the `AB` task, no combination of intra-video positive pairs or sample weights resulted in statistically significant improvements compared to dual distortion of the same image ($\delta_t = 0$). For Barlow Twins and VICReg, several nonzero δ_t resulted in significantly worse mean test AUC. Sample weights consistently made a difference in Barlow Twins across δ_t values but only improved mean test AUC for $\delta_t = 1$ and $\delta_t = 1.5$.

Different trends were observed for the `LS` task. SimCLR with $\delta_x = 5$ and no sample weights improved mean test AUC compared to the baseline where $\delta_x = 0$. No other combination of hyperparameters resulted in a significant improvement. For Barlow Twins, multiple IVPP hyperparameter combinations resulted in improved mean test AUC over the baseline. No IVPP hyperparameter combinations significantly improved the performance of VICReg.



(a) `AB` task

(b) `LS` task

Figure 5.5: Boxplots conveying the quartile ranges of test AUC distributions for each pretraining method and assignment to δ, with and without sample weights. Each experiment is repeated 20 times on disjoint subsets of the training set, each containing all images from a group of patients.

5.5 Discussion

5.5.1 Guidelines for Machine Learning Practitioners

Insights were derived to guide developers working with deep learning for ultrasound interpretation. First, SimCLR was observed to achieve the greatest performance consistently across multiple tasks. With the exception of the data-scarce COVID-19 classification task, SimCLR decisively outperformed Barlow Twins and VICReg on the `A/B` and `LS` tasks. The results provided evidence towards favouring contrastive learning over non-contrastive learning for problems in ultrasound. The results echo the observation from Chapter 3 in that SimCLR was found to be the top performing pretraining method on a local test set for `AB`. However, in Chapter 4 Barlow Twins generally outperformed SimCLR. Despite this, the M-mode positive pair sampling strategy in Chapter 4 was greatly different from this study. In Chapter 4, M-mode images were randomly selected from the same B-mode video regardless of their distance, whereas in IVPP spatial proximity is enforced. Moreover, naively sampling M-modes across the pleural line could lead to positive pairs belonging to different classes in the `LS` task in cases of lung point, which occurs when one side of the pleural line exhibits lung sliding while the other does not [93, 143].

It could be that the non-contrastive methods employed in this chapter are less effective

Pretraining	AB δ_t	AB SW	AB Mean (std) AUC	LS δ_x	LS SW	LS Mean (std) AUC
SimCLR	0	✗	0.938 (0.007)	0	✗	0.812 (0.037)
	0.5	✗	0.931 (0.010) *	5	✗	0.824 (0.030) *
	0.5	✓	0.936 (0.007) †	5	✓	0.820 (0.033)
	1	✗	0.934 (0.011)	10	✗	0.815 (0.035)
	1	✓	0.933 (0.011)	10	✓	0.816 (0.037)
	1.5	✗	0.936 (0.013)	15	✗	0.819 (0.034)
	1.5	✓	0.932 (0.012)	15	✓	0.798 (0.039) *†
Barlow Twins	0	✗	0.914 (0.014)	0	✗	0.693 (0.044)
	0.5	✗	0.914 (0.010) *	5	✗	0.694 (0.040)
	0.5	✓	0.883 (0.017) *†	5	✓	0.780 (0.040) *†
	1	✗	0.877 (0.022) *	10	✗	0.705 (0.051)
	1	✓	0.891 (0.018) *†	10	✓	0.706 (0.066)
	1.5	✗	0.870 (0.024) *	15	✗	0.769 (0.037) *
	1.5	✓	0.892 (0.015) *†	15	✓	0.707 (0.071) †
VICReg	0	✗	0.917 (0.011)	0	✗	0.690 (0.042)
	0.5	✗	0.879 (0.024) *	5	✗	0.675 (0.036)
	0.5	✓	0.879 (0.021) *	5	✓	0.679 (0.038)
	1	✗	0.872 (0.023) *	10	✗	0.680 (0.039)
	1	✓	0.876 (0.024) *	10	✓	0.675 (0.040)
	1.5	✗	0.860 (0.026) *	15	✗	0.710 (0.036)
	1.5	✓	0.870 (0.021) *†	15	✓	0.685 (0.039) †
None (ImageNet-pretrained)			0.896 (0.017)			0.783 (0.028)
None (random initialization)			0.774 (0.051)			0.507 (0.022)

* Significantly different ($p < 0.05$) than baseline for the pretraining method where $\delta = 0$

† Significantly different ($p < 0.05$) for particular δ when sample weights are applied, compared to no sample weights

Table 5.3: ParenchymalLUS test AUC for the AB and LS tasks when trained using examples from 5% of the patients in the training set. Twenty trials were performed for each pretraining method, value of δ, with and without sample weights (SW). Mean and standard deviation of the test AUC across trials are reported for each condition.

for LU examinations than SimCLR. We suspect that the lack of diversity in parenchymal lung ultrasound and the fine-grained nature of the classification tasks is problematic for non-contrastive methods, as the objectives are attractive and focus on maximizing embedding information. Perhaps explicit samples of challenging negative pairs may contribute

to learning a meaningful embedding manifold for fine-grained downstream tasks. Future work assessing non-contrastive methods for tasks in different ultrasound examinations or alternative imaging modalities altogether would shed light on the utility of non-contrastive methods outside the typical evaluation setting of photographic images.

While the experimental results did not support the existence of overarching trends for hyperparameter assignments for intra-video positive pairs across pretraining methods, it was observed that some combinations improved performance on particular downstream tasks. For example, each pretraining method's downstream performance on COVID-19 classification was improved by a nonzero value of δ_t. Overall, the results indicated that the optimal assignment for IVPP hyperparameters may be problem-specific. IVPP may improve performance on downstream ultrasound interpretation tasks; however, practitioners are advised to include a range of values of δ with and without sample weights in their hyperparameter search.

5.5.2 Limitations

The methods and experiments in this chapter were not without limitations. As is common in medical imaging datasets, the ParenchymalLUS dataset was imbalanced. The image-wise representation for the positive class was 30.0% for the `AB` task and 11.7% for the lung sliding task. Although some evidence exists in support for self-supervised pretraining for alleviating the ill effects of class imbalance in photographic images [100, 174], CT, and fundoscopy images [183], we found no such evidence for tasks in medical ultrasound.

As outlined in the background, the pretraining objectives employed in this chapter have been shown to improve downstream performance when the pairwise relationship aligns with the downstream task [11]. These guarantees compare to the baseline case of random weight initialization. While it was observed that all pretraining methods outperformed full supervision with randomly initialized weights, ImageNet-pretrained weights outperformed non-contrastive methods in several of the experiments. ImageNet-pretrained weights are a strong baseline for medical imaging tasks, as they have been shown to boost performance in several supervised learning tasks across medical imaging modalities [9]. It is possible that some extreme data augmentation transformations and intra-positive pairs could jeopardize the class agreement of positive pairs (as is likely in most pragmatic cases); however, near-consistent alignment was achieved through data augmentation design and small ranges of δ. Although there exists evidence that VICReg and SimCLR can achieve similar performance on ImageNet with judicious selection of hyperparameters (e.g., temperature, loss term weights, learning rate) [52], we used default hyperparameters. Due to limited com-

putational resources, we avoided expansion of the hyperparameter space by only studying IVPP hyperparameters.

Lastly, M-mode images were designated by selecting x-coordinates in B-mode videos that intersected a pleural line region of interest, as predicted by an object detection model utilized in previous work [156, 158]. LU M-mode images must intersect the pleural line in order to appreciate the lung sliding sign. While we mitigated potential inaccuracies in localization by limiting training and evaluation data to the brightest half of eligible x-coordinates, it is possible that a small fraction of M-mode images was utilized that did not intersect the pleural line.

5.6 Conclusion

This chapter proposed the Intra-Video Positive Pairs (IVPP) method for improving the downstream performance of ultrasound classifiers pretrained with joint embedding self supervised learning. We proposed a novel scheme for integrating such positive pairs into common contrastive and non-contrastive SSL methods. Applicable to both B-mode and M-mode ultrasound, IVPP consisted of sampling positive pairs that are separated temporally or spatially by no more than a threshold, optionally applying sample weights to each pair in the objective depending on the distance. Investigations revealed that using nearby images from the same video for positive pairs can lead to improved performance when compared to composing positive pairs from the same image, but that the best IVPP hyperparameter may vary per downstream task. Another salient result was the persistent top performance of SimCLR for key tasks in B-mode and M-mode lung ultrasound, indicating that contrastive learning may be more suitable than non-contrastive learning methods for ultrasound imaging.

Future work could investigate IVPP for other types of medical ultrasound exams. IVPP could also be integrated into other SSL objectives. The sample weights formulation proposed in this chapter could also be applied to SSL for non-ultrasound videos. Given the high performance of SimCLR, subsequent work could perform a comprehensive comparison between contrastive and non-contrastive SSL methods for other tasks in medical ultrasound.

Lastly, this chapter was focused on leveraging the video-based structure of ultrasound to enhance the positive pair sampling strategy for image-based SSL for LU data. However, recall that data augmentation is a key component of the positive pair sampling strategy in joint embedding SSL. As in prior chapters, we used data augmentation transformations drawn from a set of transformations that are commonly employed in the literature.

In the following chapter, we challenge the use of these standard transformations and explore ultrasound-specific data augmentation designed to preserve semantic content in LU images.

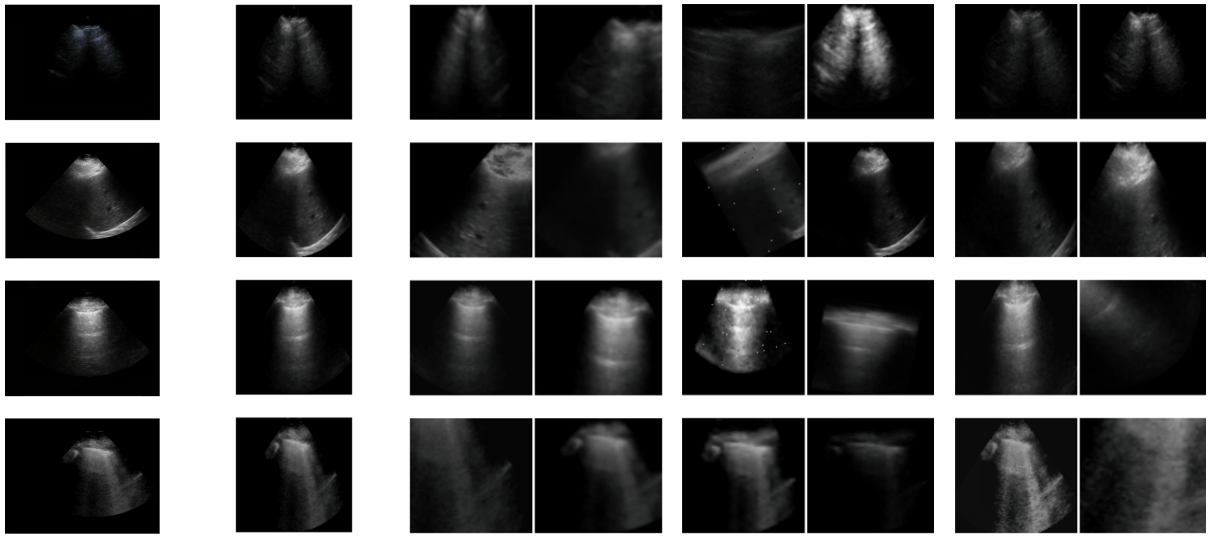
Chapter 6

The Efficacy of Semantics-Preserving Transformations in Self-Supervised Learning for Lung Ultrasound

6.1 Introduction

Joint embedding self-supervised learning methods are characterized by training DNNs to produce similar vector representations for pairs of related images. The most common method for retrieving related pairs of images from unlabelled datasets is to apply random transformations (i.e., data augmentation) to an image, producing two distorted views. The choice of random transformations steers the invariance relationships learned by the backbone. In the previous chapter, we explored a positive pair sampling scheme for ultrasound images that leverages the fact that B-mode ultrasound is collected as a stream of video frames. Pairs of proximally situated images were selected from the same video prior to applying data augmentation transformations. However, all prior chapters used data augmentation transformations that are commonly employed in the computer vision literature, which were not designed with ultrasound imaging in mind. Indeed, this is common practice when adapting self-supervised learning (SSL) methods to medical imaging in general.

This chapter presents additional methods for positive pair generation for B-mode lung ultrasound (LU) images that are characterized by domain-specific data augmentation. We challenged the notion that standard data augmentation transformations are suitable for all LU tasks. To this end, we proposed and assessed data augmentation strategies designed to preserve semantic content in medical ultrasound images. Handcrafted domain-specific



(a) Original images | (b) Preprocessed images | (c) StandardAug pipeline | (d) AugUS-O pipeline | (e) AugUS-D pipeline

Figure 6.1: Examples of the preprocessing and data augmentation methods in this chapter. **(a)** Original images are from ultrasound exams. **(b)** Semantics-preserving preprocessing is applied to crop out areas external to the field of view. **(c)** The StandardAug pipeline is a commonly employed data augmentation pipeline in self-supervised learning. **(d)** The AugUS-O pipeline was designed to preserve semantic content in ultrasound images. **(e)** AugUS-D is a hybrid pipeline whose construction was informed by empirical investigations into the StandardAug and AugUS-O pipelines.

augmentation methods were compared against standard SSL data augmentation practice. We found that ultrasound-specific transformations resulted in the greatest improvement in performance for COVID-19 classification — a diagnostic task — on a public dataset. Experiments also revealed that standard cropping-based augmentation strategies outperformed ultrasound-specific transformations for object classification tasks in LU. Lastly, we investigated the effect of ultrasound-specific preprocessing methods during self-supervised pretraining. Ultrasound-specific semantics-preserving preprocessing was found to be instrumental to the success of pretrained backbones. In summary, our contributions are as follows:

- An assessment of semantics-preserving image preprocessing for SSL with ultrasound images

- Semantics-preserving data augmentation methods designed for SSL with ultrasound images
- Comparison of multiple data augmentation strategies for SSL for multiple types of LU tasks
- Recommendations for machine learning practitioners working with unlabelled ultrasound datasets

Figure 6.1 summarizes the methods that were assessed in this chapter. To our knowledge, this chapter constitutes the first attempt to quantify the impact of data augmentation methods for SSL with ultrasound.

6.2 Background

6.2.1 Data Augmentation in Self-Supervised Learning

The joint embedding class of SSL methods is characterized by the minimization of an objective function that, broadly speaking, encourages similarity of related pairs of inputs. Semantically related pairs of images (i.e., *positive pairs*) are sampled from unlabelled datasets according to a *pairwise relationship*. If the SSL pairwise relationship is satisfied for samples exhibiting the same class, SSL methods will likely improve the performance of a classifier [11]. Most joint embedding methods rely on data augmentation to define the pairwise relationship. Some studies have used meta-data or known relationships between samples to identify related pairs [9,13,189]; however, the availability of such information is rare. The choice of data augmentation transformations is therefore crucial, as it dictates the invariance relationships learned [23]. However, the set of useful invariances differs by the image modality and downstream problem(s) of interest. Despite this, studies continue to espouse a data augmentation pipeline popularized by leading SSL methods, such as SimCLR [29], BYOL [59], Barlow Twins [180], and VICReg [12]. These methods utilized the same pipeline, but with minor hyperparameter variations. The pipeline includes the following transformations: random crops, horizontal reflection, colour jitter, Gaussian blur, and solarization. Hereafter, we refer to this baseline pipeline as *StandardAug*. Random rotation is an example of a transformation not found in the StandardAug pipeline that represents an important invariance relationship for many tasks in medical imaging. For example, random rotation has been applied in SSL pretraining with magnetic resonance exams of the prostate [46]. Moreover, the authors did not use StandardAug's Gaussian blur transformation because it may have rendered the images uninterpretable.

6.2.2 Data Augmentation for Self-Supervision in Ultrasound

Recent studies have examined the use of joint embedding SSL methods for ultrasound interpretation tasks, such as echocardiogram view classification [4], left ventricle segmentation [132], and breast tumour classification [118]. Some have proposed positive pair sampling schemes customized for ultrasound. The USCL and Meta-USCL methods explored contrastive learning techniques where the positive pairs were weighted sums of images from the same ultrasound video that were then subjected to data augmentation [32, 33]. Other methods have studied the use of images from the same video as positive pairs [13, 182], including the method introduced in Chapter 5 [157]. In these studies, the set of transformations were a subset of the StandardAug data augmentation pipeline, occasionally with different hyperparameters. Few studies have proposed ultrasound-specific data augmentation methods for SSL. A recent study by Chen *et al.* [28] applied BYOL and SimCLR to pretrain 3D CNNs with specialized data augmentation for lung consolidation detection in ultrasound videos, observing that temporal transformations were contributory to their problem. This chapter builds on previous literature by proposing and comparing domain-specific data augmentation and preprocessing methods for multiple types of downstream tasks.

6.3 Methods

6.3.1 Datasets and Tasks

We assessed the methods proposed in this chapter using a combination of public and private data. COVIDx-US is a public COVID-19 LU dataset consisting of 242 publicly sourced videos, acquired from a variety of manufacturers and sites [43]. Each example is annotated with one of the following classes: normal, COVID-19 pneumonia, non-COVID-19 pneumonia, and other lung pathology. Referred to as `COVID` hereafter, the task is a four-class image classification problem. Since there is no standard test partition, we split the data by patient identifier into training (70%), validation (15%), and test (15%) splits. Table 6.1 shows how acquisition characteristics are distributed over COVIDx-US.

The second data source was a private collection of lung ultrasound (LU) examinations, and we refer to it as *LUSData*. Appendix A elaborates on how access to the data was granted. LUSData contains videos from parenchymal and pleural views of the lung. A subset of the parenchymal views has labels for the presence of A-lines or B-lines (i.e., the `AB` classification task). A-lines are reverberation artefacts that indicate normal lung tissue,

	Train	Validation	Test
Probe Type			
Phased Array	55 (32.5%)	18 (42.9%)	11 (35.5%)
Curved Linear	83 (49.1%)	13 (31.0%)	10 (32.3%)
Linear	31 (18.3%)	11 (26.2%)	10 (32.3%)
Patient Sex			
Male	33 (19.5%)	15 (35.7%)	7 (22.6%)
Female	18 (10.7%)	3 (7.1%)	3 (9.7%)
Unknown	118 (69.8%)	24 (57.1%)	21 (67.7%)
Patient Age			
Mean [STD]	36.6 [14.6]	47.6 [18.9]	52.9 [21.1]
Unknown	127 (75.1%)	22 (52.4%)	20 (64.5%)
Total	169	42	31

Table 6.1: Characteristics of the ultrasound videos in the COVIDx-US dataset. The number of videos possessing known values for a variety of attributes are displayed. Percentages of the total in each split are provided as well, but some do not sum to 100 due to rounding.

while B-lines are axial artefacts that indicate fluid or thickness in the lung. A subset of the pleural views is labelled for the presence or absence of pleural effusions (i.e., the `PE` classification task), which is an accumulation of fluid around the lungs. A small fraction of the parenchymal views in LUSData possess bounding box labels for the pleural line (i.e., the `PL` object detection task). Most exams in LUSData originated from a local healthcare centre, but a subset were acquired at another centre, which we adopt as an external test set. The labelled examples in the local dataset were split into training (70%), validation (15%), and test (15%) splits by patient. Table 6.2 provides the video and class counts of LUSData. All models in this chapter were trained on images, instead of on videos. Classification labels apply to every image in the video. However, individual images within each video labelled for the `PL` task have bounding box annotations. Table 6.3 provides a breakdown of LUSData by acquisition characteristics. The external test set's manufacturer and probe type distributions differ greatly from those in the local test set. As such, the external test set constitutes a meaningful assessment of generalizability for models trained using the LUSData dataset.

	Local				External
	Unlabelled	Train	Validation	Test	Test
Patients	5571	1702	364	364	168
Videos	59 309	5679	1184	1249	925
Images	1.3×10^7	1.2×10^6	2.5×10^5	2.6×10^5	1.1×10^5
`AB` labels	N/A	2067 / 999	459 / 178	458 / 221	286 / 327
`PE` labels	N/A	789 / 762	176 / 142	162 / 158	68 / 110
`PL` labels	N/A	200	39	45	0

Table 6.2: Breakdown of the unlabelled, training, validation, and test sets in the private dataset. For each split, we indicate the number of distinct patients, videos, and images. x/y indicates the number of labelled videos in the negative and positive class for each binary classification task. For the `PL` task, we indicate the number of videos with frame-level bounding box annotations.

6.3.2 Semantics-Preserving Preprocessing

The field of view (FOV) in ultrasound images is typically surrounded by burnt-in scan parameters, logos, and other details. We estimated the shape of the FOV and redacted all extraneous graphical entities using ultrasound cleaning software (UltraMask, Deep Breathe Inc., London, ON, Canada). UltraMask also classifies the probe type in each ultrasound video using its estimated FOV shape, which was how we determined the probe type for each example in LUSData and COVIDx-US (see Tables 6.3 and 6.1).

Semantic information only exists within the ultrasound FOV, which typically occupies a fraction of the images. Scaling transformations, such as random cropping, could produce views that largely contain background. Accordingly, we cropped the cleaned images to the smallest rectangle that encapsulates the FOV shape to maximize semantic content in ultrasound images. Figure 6.2 depicts this semantics-preserving preprocessing workflow. The process was applied to all images in LUSData and COVIDx-US.

6.3.3 Ultrasound-specific Data Augmentation

Joint embedding SSL is effective when positive pairs contain similar information with respect to downstream tasks [11]. Several SSL studies applied to photographic or medical imaging datasets adopted variations of the StandardAug data augmentation pipeline. The

	Local				External
	Unlabelled	Train	Validation	Test	Test
Probe Type					
Phased Array	50 769 (85.6%)	5146 (90.6%)	1051 (88.8%)	1062 (85.0%)	586 (63.4%)
Curved Linear	6601 (11.1%)	439 (7.7%)	108 (9.1%)	167 (13.4%)	92 (9.9%)
Linear	1939 (3.3%)	94 (1.7%)	25 (2.1%)	20 (1.6%)	247 (26.7%)
Manufacturer					
Sonosite	53 663 (90.5%)	4386 (77.2%)	848 (71.6%)	963 (77.1%)	626 (67.7%)
Mindray	4045 (6.8%)	847 (14.9%)	216 (18.2%)	153 (12.2%)	55 (5.9%)
Philips	66 (0.1%)	50 (0.9%)	6 (0.5%)	11 (0.9%)	244 (26.4%)
Esaote	233 (0.4%)	4 (0.1%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
General Electric	10 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
Depth (cm)					
Mean [STD]	14.3 [4.5]	13.0 [3.8]	13.1 [3.7]	12.7 [3.8]	11.1 [4.8]
Unknown	1606 (2.7%)	407 (7.2%)	115 (9.7%)	122 (9.8%)	0 (0.0%)
Environment					
Intensive Care	43 839 (73.9%)	3722 (65.5%)	706 (59.6%)	727 (58.2%)	0 (0.0%)
Emergency Room	13 280 (22.4%)	760 (13.4%)	206 (17.4%)	253 (20.3%)	0 (0.0%)
Ward	2033 (3.4%)	173 (3.0%)	25 (2.1%)	49 (3.9%)	0 (0.0%)
Urgent Care	129 (0.2%)	3 (0.1%)	0 (0.0%)	1 (0.1%)	0 (0.0%)
Unknown	28 (0.0%)	1021 (18.0%)	247 (20.9%)	219 (17.5%)	925 (100.0%)
Patient Sex					
Male	30 300 (51.1%)	2963 (52.2%)	607 (51.3%)	588 (47.1%)	0 (0.0%)
Female	20 809 (35.1%)	1793 (31.6%)	325 (27.4%)	412 (33.0%)	0 (0.0%)
Unknown	8200 (13.8%)	923 (16.3%)	252 (21.3%)	249 (19.9%)	925 (100.0%)
Patient Age					
Mean [STD]	62.3 [20.0]	63.3 [16.5]	62.0 [18.4]	62.8 [17.3]	–
Unknown	53 (0.1%)	1029 (18.2%)	247 (20.9%)	219 (17.5%)	925 (100.0%)
Total	59 309	5679	1184	1249	925

Table 6.3: Characteristics of the ultrasound videos in the LUSData dataset. The number of videos possessing known values for a variety of attributes are displayed. Percentages of the total in each split are provided as well, but some do not sum to 100 due to rounding.

core aim of our study was to determine if semantics-preserving data augmentation would better equip pretrained feature extractors for downstream LU tasks than the commonly applied StandardAug pipeline.

We refer to a *data augmentation pipeline* as an ordered sequence of transformations,

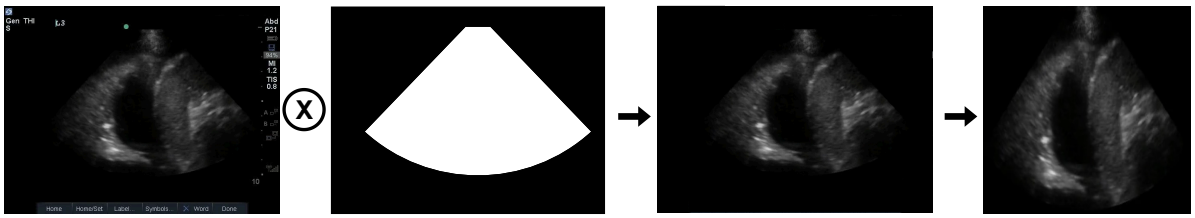


Figure 6.2: Raw ultrasound images are preprocessed by applying a mask that preserves only the field of view. Then, they are cropped according to the bounds of the field of view and resized to the backbone's expected input shape.

each applied with some probability. For clarity, we assign each transformation an alphanumeric identifier and express a data augmentation pipeline as an ordered sequence of identifiers. The StandardAug pipeline transformations are detailed in Table 6.4. The table also includes an estimate of the time to transform a single image. Estimates were calculated by conducting the transformation 1000 times using the same image, using a computer running Windows equipped with an Intel i9-10900K CPU at 3.7 GHz. Python 3.11 was utilized, and the transforms were written using PyTorch version 2.2.1 and TorchVision 0.17.1. Note that runtimes may vary considerably depending on the software environment and underlying hardware.

Identifier	Probability	Transformation	Time [ms]
B00	1.0	Crop and resize	0.29
B01	0.5	Horizontal reflection	0.08
B02	0.8	Colour jitter.	2.40
B03	0.2	Conversion to grayscale	0.19
B04	0.5	Gaussian blur	0.74
B05	0.1	Solarization	0.15

Table 6.4: The sequence of transformations in the StandardAug data augmentation pipeline

We designed the *AugUS-O* pipeline, which was intended to preserve semantic information in the entire ultrasound FOV while imposing nontrivial differences across invocations. The transformations in AugUS-O are listed below.

U00: **Probe Type Change:** Inspired by Zeng *et al.*'s work [181], this transformation resamples an ultrasound image according to a different FOV shape. Linear FOV

shapes are converted to curvilinear shapes, while curvilinear and phased array are converted to linear.

U01: **Convexity Change:** The shape of convex beams can vary, depending on the manufacturer, depth, and field of view of the probe. This transformation modifies the FOV shape such that the distance between the top two corners is altered, mimicking a change in the angle of view.

U02: **Wavelet Denoising:** As an alternative to the commonly used Gaussian blur transformation, this transformation denoises an image by thresholding it in wavelet space, according to Birgé and Massart's method [15].

U03: **Contrast-Limited Adaptive Histogram Equalization:** This transformation enhances contrast by applying locally informed equalization [127].

U04: **Gamma Correction:** In contrast to standard brightness change transforms, gamma correction applies a nonlinear change in pixel intensity.

U05: **Brightness and Contrast Change:** The brightness and contrast of the image are modified by applying a linear transform to the pixel values.

U06: **Depth Change Simulation:** Changing the depth controls on an ultrasound probe impacts how far the range of visibility is from the probe. This transformation simulates a change in depth by applying a random zoom, while preserving the FOV shape.

U07: **Speckle Noise Simulation:** Speckle noise, Gaussian noise, and salt & pepper noise are prevalent in ultrasound [160]. This transformation applies Singh *et al.*'s [142] synthetic speckle noise algorithm to the image.

U08: **Gaussian Noise Simulation:** Multiplicative Gaussian noise is independently applied to each pixel.

U09: **Salt & Pepper Noise Simulation:** A small, random assortment of pixels are set to black or white.

U10: **Horizontal Reflection:** The image is reflected about the central vertical.

U11: **Rotation & Shift:** The image is rotated and translated by a random angle and vector, respectively.

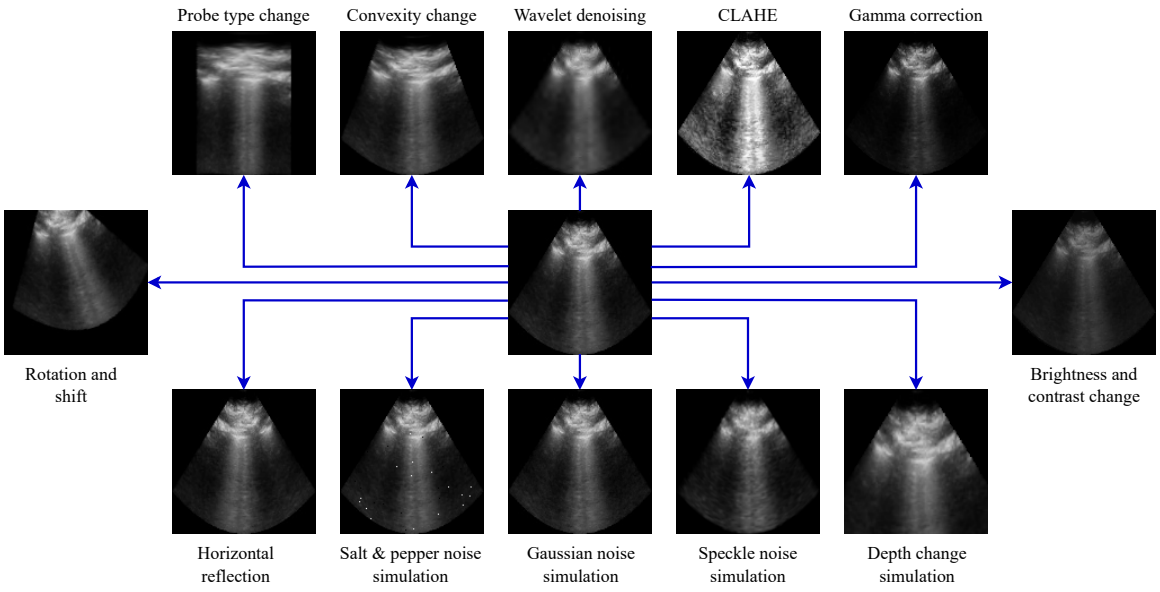


Figure 6.3: Examples of data augmentation transformations from the AugUS-O pipeline applied to the same lung ultrasound image.

Refer to Figure 6.3 for a visual example of each transformation in AugUS-O. Algorithmic details and parameter settings for the StandardAug and AugUS-O pipelines are in Appendices F.1 and F.2, respectively. As is common in stochastic data augmentation, each transformation was applied with some probability. Table 6.5 gives the entire sequence of transformations, the probability with which each is applied. Visuals of positive pairs produced using the StandardAug and AugUS-O augmentation pipelines can be found in Figure 6.4a and Figure 6.4b, respectively.

We conducted an informal assessment of the similarity of positive pairs. Positive pairs were produced for 50 randomly sampled images, using both the StandardAug and the AugUS-O pipelines. The pairs were presented in random order to an expert in point-of-care ultrasound. They were aware of the two pipelines but were not told which pipeline produced each pair. The expert was asked to mark the pairs they believed conveyed the same clinical impression. We observed that 58% of pairs produced with the StandardAug pipeline were marked as similar, whereas 70% of the AugUS-O pairs were marked as similar. While not conclusive, this manual evaluation added credence to the semantics-preserving intention of the design.

Identifier	Probability	Transformation	Time [ms]
U00	0.3	Probe type change	2.25
U01	0.75	Convexity change	1.92
U02	0.5	Wavelet denoising	5.00
U03	0.2	CLAHE†	4.64
U04	0.5	Gamma correction	0.52
U05	0.5	Brightness and contrast change	0.49
U06	0.5	Depth change simulation	1.76
U07	0.333	Speckle noise simulation	3.69
U08	0.333	Gaussian noise simulation	0.28
U09	0.1	Salt & pepper noise simulation	0.18
U10	0.5	Horizontal reflection	0.19
U11	0.5	Rotation & shift	1.42

† Contrast-limited adaptive histogram equalization

Table 6.5: The sequence of transformations in the ultrasound-specific augmentation pipeline

6.3.4 Discovering Semantically Contributory Transformations

A major aim of this investigation was to explore the utility of various data augmentation schemes during pretraining. As such, we conducted leave-one-out analysis for each of the StandardAug and AugUS-O pipelines to estimate the impact of each transformation on models' ability to solve downstream classification tasks. We pretrained separate models on the unlabelled portion of LUSData, using an altered version of a pipeline with one transformation omitted. We then conducted 10-fold cross-validation on the LUSData training set for downstream classification tasks for each pretrained model. The median cross-validation test performance for each model pretrained using an ablated pipeline was compared to a baseline model that was pretrained with the entire pipeline. The experiment was conducted for both the StandardAug and AugUS-O pipelines. Any transformations that, when omitted, resulted in worsened performance on either `AB` or `PE` were deemed contributory.

6.3.5 Training Protocols

We adopted the MobileNetV3Small architecture [71] for all experiments in this chapter and pretrained using the SimCLR method [29]. MobileNetV3Small was chosen due to its real-time inference capability on mobile devices and its use in prior work by VanBerlo *et al.* for similar tasks [155]. The SimCLR projector was a 2-layer MLP with 576 nodes per layer. Images were resized to 128×128 pixels prior to the forward pass. Unless otherwise stated, backbones were initialized using ImageNet-pretrained weights [35] and were pretrained using the LARS optimizer [177] with a batch size of 1024, a base learning rate of 0.2 and a linear warmup with cosine decay schedule. Pretraining was conducted for 3 epochs with 0.3 warmup epochs for LUSData, and 100 epochs with 10 warmup epochs for COVIDx-US.

To conduct supervised evaluation, a perceptron classification head was appended to the final pooling layer of the backbone. Classifiers were trained using stochastic gradient descent with a momentum of 0.9 and a batch size of 512. The learning rates for the backbone and head were 0.002 and 0.02, respectively; each was annealed according to a cosine decay schedule. Training was conducted for 10 epochs on LUSData and 30 epochs on COVIDx-US. Unless otherwise stated, the weights corresponding to the epoch with the lowest validation loss were retained for test set evaluation.

Although this chapter focused on classification tasks, we also evaluated backbones on the `PL` object detection task using the Single Shot Detector (SSD) method [103]. SSL-pretrained backbones were used as the convolutional feature extractor. Architectural and training details for SSD are in Appendix G.

Self-supervised pretraining was conducted using virtual machines equipped with an Intel E5-2683 v4 Broadwell CPU at 2.1 GHz and 2 Nvidia Tesla P100 GPUs each with 12 GB of VRAM. Supervised training was conducted using the same hardware, except with a single GPU. Source code for the experiments and transformations is available in a public GitHub repository.[1]

6.4 Experiments and Results

6.4.1 Transformation Leave-one-out Analysis

A leave-one-out analysis was conducted to discover which transformations in each of the StandardAug and AugUS-O pipelines were contributory to downstream task performance.

[1] https://github.com/bvanberl/aug_us_ssl

We pretrained backbones using versions of each pipeline with one transformation omitted. The private LUSData training set was split by patient into 10 disjoint subsets. For each pretrained backbone, 10-fold cross-validation was conducted to obtain estimates of the performance of linear classifiers trained on its output feature vectors. The maximum validation area under the receiver operating characteristic curve (AUC) across epochs was recorded. Omitted transformations that resulted in statistically significant lower validation AUC for either the `AB` or `PE` task were included in a third pipeline.

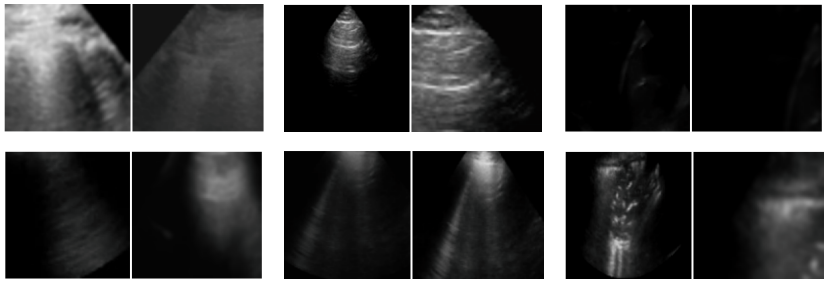
We conducted statistical testing to compare each of the StandardAug and AugUS-O pipelines, and for each of the `AB` and `PE` tasks (described in Section 6.3.1). Friedman's Test for multiple comparisons [48] was conducted, with significance level 0.05. When significant differences were found, we performed the Wilcoxon Sign-Rank Test [165] to compare the test AUCs from each ablated model to the baseline's test AUCs. To control for false positives, the Holm-Bonferroni correction [70] was applied to keep the family-wise significance level at 0.05.

Table 6.6 details the results of the leave-one-out analysis. Friedman's Test detected differences in performance on both the `AB` and `PE` tasks when pretrained using the StandardAug pipeline, but only the `AB` task exhibited differences when pretrained with the AugUS-O pipeline. As shown in Table 6.6, the set of transformations that exhibited statistically significant reductions in test AUC for at least one task when excluded were crop & resize (B00), colour jitter (B02), CLAHE (U03), and rotation & shift (U11). Appendix H provides all test statistics from this investigation. Of note is the sharp decrease in performance without the random crop and resize (B00), indicating that it is a critical transformation.

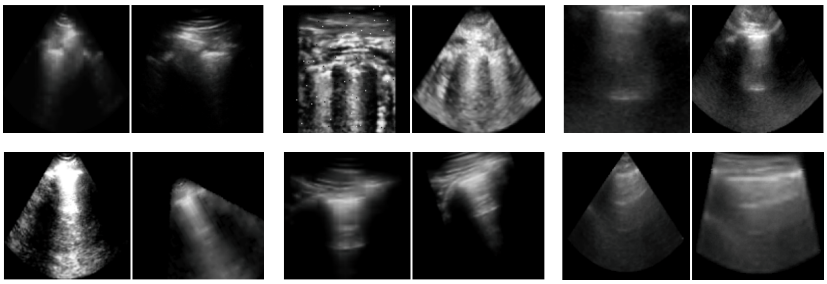
Using these transformations, we constructed a distilled pipeline that consists only of the above transformations. Referred hereafter to as *AugUS-D*, the pipeline is expressed as the following sequence: *[U03, B02, U11, B00]*. Figure 6.4c provides some examples of positive pairs produced with AugUS-D. For more examples of pairs produced by each pipeline, see Appendix I.

6.4.2 Object Classification Task Evaluation

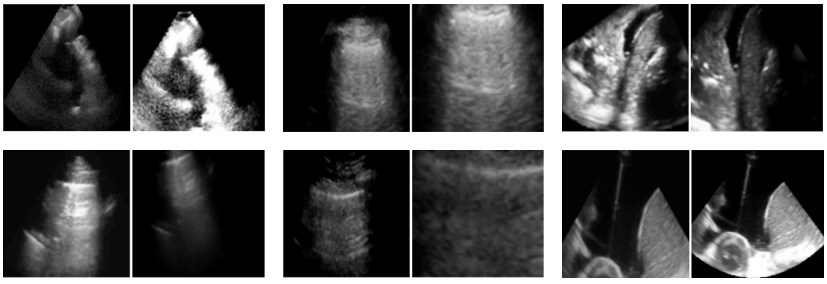
The StandardAug, AugUS-O, and AugUS-D pipelines were compared in terms of their performance on multiple downstream tasks. Model backbones were pretrained using each of the data augmentation pipelines on the union of the unlabelled and training sets in LUSData. Linear evaluation and fine-tuning experiments were performed according to the procedure explained in Section 6.3.5. In this section, we present results on the two object



(a) StandardAug pipeline



(b) AugUS-O pipeline



(c) AugUS-D pipeline

Figure 6.4: Examples of positive pairs produced using each of the StandardAug, AugUS-O, and AugUS-D data augmentation pipelines.

Pipeline	Omitted	AB		PE	
		Mean (std)	Median	Mean	Median
StandardAug	None	0.978 (0.007)	0.978	0.852 (0.040)	0.845
	B00	0.864 (0.022)	0.873†	0.695 (0.050)	0.707†
	B01	0.976 (0.006)	0.974	0.848 (0.046)	0.856
	B02	0.975 (0.007)	0.975†	0.840 (0.046)	0.842†
	B03	0.978 (0.007)	0.978	0.849 (0.044)	0.846
	B04	0.976 (0.007)	0.975	0.840 (0.041)	0.842
	B05	0.977 (0.007)	0.977	0.851 (0.041)	0.853
AugUS-O	None	0.956 (0.013)	0.959	0.828 (0.030)	0.837
	U00	0.958 (0.011)	0.957	0.831 (0.034)	0.839
	U01	0.952 (0.016)	0.952	0.835 (0.027)	0.838
	U02	0.965 (0.011)	0.967§	0.840 (0.032)	0.851
	U03	0.950 (0.011)	0.951†	0.825 (0.028)	0.827
	U04	0.957 (0.013)	0.958	0.831 (0.034)	0.836
	U05	0.953 (0.014)	0.952	0.839 (0.024)	0.845
	U06	0.961 (0.009)	0.959	0.829 (0.037)	0.833
	U07	0.959 (0.012)	0.960	0.838 (0.035)	0.856
	U08	0.961 (0.013)	0.966§	0.834 (0.027)	0.849
	U09	0.962 (0.012)	0.967§	0.838 (0.030)	0.845
	U10	0.956 (0.011)	0.959	0.826 (0.035)	0.838
	U11	0.937 (0.020)	0.939†	0.825 (0.028)	0.823

† Median is significantly less than baseline, where no transformations were omitted.
§ Median is significantly greater than baseline, where no transformations were omitted.

Table 6.6: A comparison of ablated versions of the StandardAug and AugUS-O pipeline with one excluded transformation versus the original pipelines. Models were pretrained on the LUSData unlabeled set and evaluated on two downstream classification tasks – AB and PE. Performance is expressed as mean and median test AUC from 10-fold cross-validation achieved by a linear classifier trained on the feature vectors of a frozen backbone.

classification tasks: A-line vs B-line classification (AB) and pleural effusion classification (PE).

Linear classifiers indicate the usefulness of pretrained backbones, as the only trainable weights for supervised learning are those belonging to the perceptron head. Table 6.7 reports the test set performance of linear classifiers for each task and data augmentation

pipeline. On the private dataset, the AugUS-D and StandardAug pipelines performed comparably well on the AB task. AugUS-D attained greater performance metrics than the StandardAug pipeline on PE. To provide a visual perspective on linear classifier performance, we produced two-dimensional t-SNE embeddings of the feature vectors outputted by pretrained backbones for examples in the test set [110]. Shown in Figure 6.5, the separability of the visual representations is consistent with linear classifier performance.

Train Setting	Task	Initial Weights	Pipeline	Accuracy	Precision	Recall	AUC
LC	AB	SimCLR	StandardAug	**0.932**	**0.951**	0.819	0.970
		SimCLR	AugUS-O	0.910	0.939	0.756	0.953
		SimCLR	AugUS-D	0.931	0.947	**0.820**	**0.971**
		ImageNet	-	0.898	0.894	0.758	0.949
	PE	SimCLR	StandardAug	0.782	0.769	0.787	0.881
		SimCLR	AugUS-O	0.795	0.796	0.776	0.853
		SimCLR	AugUS-D	**0.800**	**0.798**	0.782	**0.886**
		ImageNet	-	0.779	0.756	**0.804**	0.864
FT	AB	SimCLR	StandardAug	**0.941**	0.951	0.850	**0.970**
		SimCLR	AugUS-O	0.939	0.938	**0.859**	0.968
		SimCLR	AugUS-D	0.931	**0.960**	0.809	0.962
		Random	-	0.883	0.794	0.834	0.938
		ImageNet	-	0.911	0.872	0.830	0.953
	PE	SimCLR	StandardAug	0.766	0.713	0.863	0.882
		SimCLR	AugUS-O	0.487	0.479	0.685	0.557
		SimCLR	AugUS-D	**0.802**	0.782	0.818	**0.884**
		Random	-	0.703	**0.733**	0.607	0.767
		ImageNet	-	0.708	0.640	**0.907**	0.845

Table 6.7: Test set performance for linear classification (LC) and fine-tuning (FT) experiments with the AB and PE tasks. Binary metrics are averages across classes. The best observed metrics in each experimental setting are in **boldface**.

We fine-tuned the pretrained models, allowing the backbone's weights to be trainable in addition to the model head. Table 6.7 gives the test set performance of the fine-tuned classifiers. We observed similar performance differences among the different augmentation pipelines but note some additional findings. The model pretrained using AugUS-O on LUSData performed comparably against the other pipelines on AB but exhibited extremely

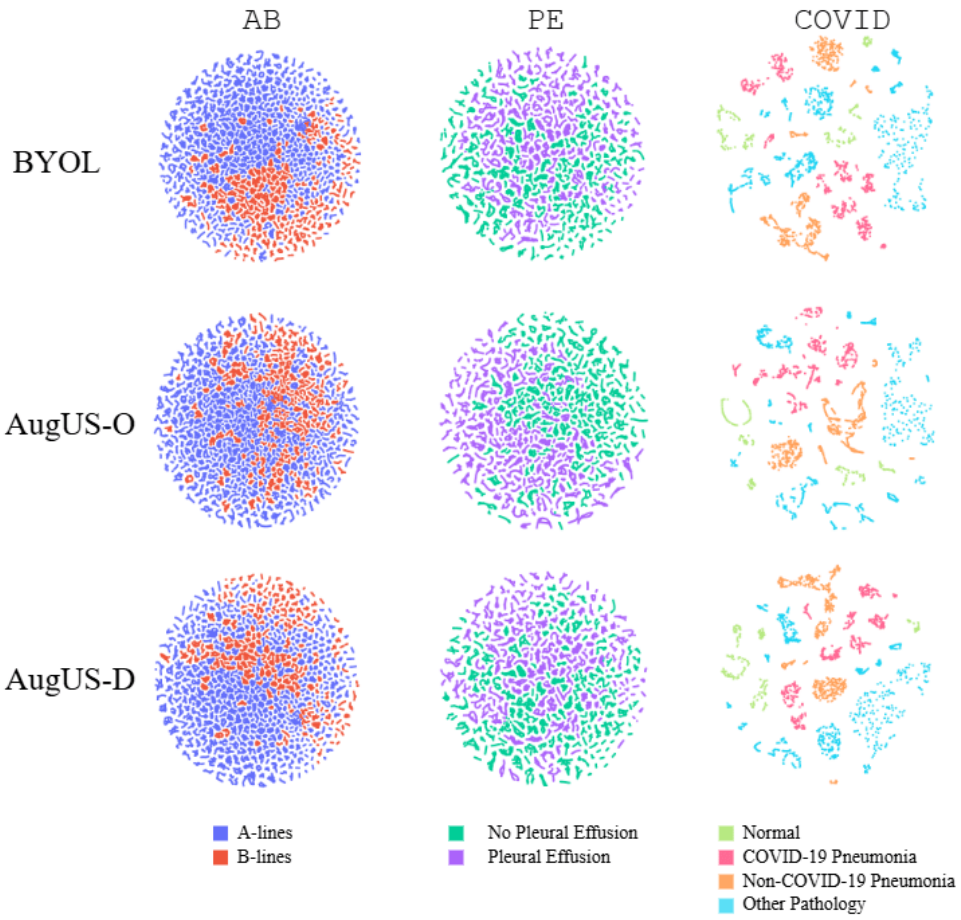


Figure 6.5: Two-dimensional t-SNE projections for test set feature vectors produced by SimCLR-pretrained backbones, for all tasks and data augmentation pipelines.

poor performance on the `PE` test set. Although it may appear that this model may have overfit to the training set, examination of training metrics revealed that training and validation metrics were close, with validation set AUC having been evaluated as 0.861. Nonetheless, fine-tuned models that were pretrained with the StandardAug and AugUS-O pipelines yielded strong performance on both tasks in LUSData.

Linear and fine-tuned classifiers for the `AB` and `PE` tasks were also evaluated on the external LUSData test set. Metrics for external test set predictions are provided in Table 6.8. Most classifiers exhibited degraded performance on external data, compared to the local test set. Overall, the relative performance of the classifiers on the external test

set was reflective of their performance on local test data. As was observed in Chapter 3, SimCLR-pretrained linear and fine-tuned classifiers achieved greater performance than those initialized with ImageNet-pretrained or randomly sampled weights on external AB and PE test sets. Note the difference in the manufacturer and probe type distributions between the local and external datasets (see Table 6.3) — there was a far greater proportion of videos gathered by linear probes and Philips-manufactured devices in the external test set. Unlike the local test evaluation, the network trained from scratch performed comparably on AB to the SimCLR-pretrained models that utilized the AugUS-O and AugUS-D pipelines. However, the SimCLR-pretrained model that utilized the StandardAug pipeline achieved the greatest AB test AUC by a margin of 0.029 among the fine-tuned models. On the PE task, the classifier originating from the same pretrained model that utilized StandardAug achieved the greatest AUC by a margin of 0.019 among the fine-tuned models. In the linear classifier setting, models pretrained using StandardAug and AugUS-D performed comparably well on the external test set, achieving greater AUC than the model pretrained using AugUS-O. Similar to the local test set, the pretrained models that incorporated the StandardAug and AugUS-D pipeline achieved the greatest test AUC, while the pretrained model that utilized AugUS-O performed the worst.

Although MobileNetV3Small was the backbone architecture used for these experiments, we repeated the above evaluations using the more commonly employed ResNet18 architecture [68]. Similar trends were observed regarding the greater test performance attained by models pretrained with cropping-based pipelines. However, the fine-tuned models greatly overfit, likely due to ResNet18's much greater capacity. The ResNet18 models that achieved the greatest test performance were mostly the linear classifiers trained on frozen backbones. Notably, the trend persisted when evaluating on external test data. Detailed results for ResNet18 can be found in Appendix J.

6.4.3 Diagnostic Classification Task Evaluation

Models pretrained on LUSData were also evaluated on the COVID-19 multi-class problem (COVID). Unlike the AB and PE tasks, COVID is a diagnostic task that involves global image understanding, as the relationship between objects is pertinent. Multiple findings on lung ultrasound have been observed in the context of COVID-19 pneumonia, including B-lines, pleural line abnormalities, and consolidation [16].

Linear classifiers were trained on the COVID training set and evaluated on the COVID test set. As shown in Table 6.9, AugUS-O was observed to have the greatest test multi-class AUC, which was considered the primary metric of interest. Looking at the t-SNE

Train Setting	Task	Initial Weights	Pipeline	Accuracy	Precision	Recall	AUC
LC	AB	SimCLR	StandardAug	**0.749**	**0.956**	**0.555**	**0.868**
		SimCLR	AugUS-O	0.689	0.927	0.453	0.810
		SimCLR	AugUS-D	0.726	0.922	0.531	0.859
		ImageNet	-	0.643	0.872	0.389	0.770
	PE	SimCLR	StandardAug	0.794	0.835	**0.843**	0.880
		SimCLR	AugUS-O	0.784	**0.916**	0.728	0.870
		SimCLR	AugUS-D	**0.806**	0.877	0.809	**0.887**
		ImageNet	-	0.758	0.836	0.771	0.840
FT	AB	SimCLR	StandardAug	**0.751**	**0.961**	0.556	**0.883**
		SimCLR	AugUS-O	0.734	0.960	0.523	0.850
		SimCLR	AugUS-D	0.712	0.934	0.500	0.854
		Random	-	0.748	0.885	**0.606**	0.853
		ImageNet	-	0.718	0.903	0.527	0.814
	PE	SimCLR	StandardAug	**0.805**	0.838	0.861	**0.898**
		SimCLR	AugUS-O	0.572	0.649	0.717	0.536
		SimCLR	AugUS-D	0.800	**0.904**	0.768	0.879
		Random	-	0.700	0.850	0.643	0.804
		ImageNet	-	0.776	0.775	**0.914**	0.840

Table 6.8: External test set metrics for linear classification (LC) and fine-tuning (FT) experiments with the AB and PE tasks. Binary metrics are averages across classes. The best observed metrics in each experimental setting are in **boldface**.

visualizations in Figure 6.5, AugUS-O corresponds to the only visualization where the representations for the COVID-19 Pneumonia and non-COVID-19 Pneumonia classes are each clustered together.

Table 6.9 provides test metrics for fine-tuned COVID classifiers. Again, AugUS-O exhibited the best performance. Moreover, fine-tuned models generally performed worse than the linear classifiers trained on feature vectors from SSL-pretrained models; they likely suffered from overfitting, as COVIDx-US is a smaller dataset.

Unlike the AB and PE tasks, models trained for COVID were pretrained using the LUSData dataset. We repeated the linear classification and fine-tuning experiments using models pretrained on the COVIDx-US training set. Table 6.9 reports the results of linear and fine-tuning evaluations. As was observed for backbones pretrained on LUSData,

pretraining with the AugUS-O pipeline resulted in the greatest test set AUC.

The trends observed for the `COVID` task are different than those observed for the `AB` and `PE` tasks. Regardless of the augmentation pipeline, SimCLR-pretrained weights resulted in better performance than ImageNet-pretrained or random weight initialization. On object classification tasks, models pretrained using the StandardAug and AugUS-D pipelines performed the best. However, on the diagnostic `COVID` task, AugUS-O performed best. Recall that AugUS-O was designed to retain semantic information, while both StandardAug and AugUS-D contain the very impactful crop and resize (C&R) transform that can obscure substantial portions of the image. Object classification tasks require scale invariance, which is enforced by applying FOV during SSL pretraining. Diagnostic tasks, on the other hand, require global image context for interpreters to make a decision, which is preserved best by the AugUS-O pipeline.

Train Setting	Pretraining Dataset	Initial Weights	Pipeline	Accuracy	Precision	Recall	AUC
LC	LUSData	SimCLR	StandardAug	0.454	0.371	0.413	0.784
		SimCLR	AugUS-O	**0.560**	0.431	**0.513**	**0.836**
		SimCLR	AugUS-D	0.487	0.348	0.447	0.713
	COVIDx-US	SimCLR	StandardAug	0.498	**0.582**	0.501	0.781
		SimCLR	AugUS-O	**0.557**	0.506	**0.543**	**0.820**
		SimCLR	AugUS-D	0.540	0.400	0.491	0.760
	-	ImageNet	-	0.503	0.304	0.451	0.699
FT	LUSData	SimCLR	StandardAug	0.381	0.259	0.365	0.753
		SimCLR	AugUS-O	**0.557**	**0.428**	**0.509**	**0.836**
		SimCLR	AugUS-D	0.465	0.321	0.430	0.744
	COVIDx-US	SimCLR	StandardAug	0.450	**0.540**	0.464	0.770
		SimCLR	AugUS-O	0.517	0.483	**0.510**	**0.814**
		SimCLR	AugUS-D	**0.526**	0.384	0.479	0.672
	-	Random	-	0.423	0.327	0.401	0.534
	-	ImageNet	-	0.502	0.305	0.457	0.698

Table 6.9: Test set performance for linear classification (LC) and fine-tuning (FT) experiments with the `COVID` task. Binary metrics are averages across classes. The best observed metrics in each experimental setting are in **boldface**.

6.4.4 Object Detection Task Evaluation

Recall that the PL task is an object detection problem geared toward localizing the pleural line. We evaluated the pretrained models on PL to explore whether the trends observed for object-centric lung US classification tasks would hold for an object detection task, where locality understanding is explicit. We considered two evaluation settings: one in which the pretrained backbones weights were held constant, and another in which the backbone's weights were trainable. Table 6.10 reports the average precision at a 50% intersection over union threshold (AP@50) evaluated on the LUSData test set. When the backbone weights were frozen, SimCLR pretraining with AugUS-O resulted in the greatest test AP@50. These trends differed from the results observed for AB and PE classification, which both require object recognition. We speculate that aggressive cropping during the pretraining phase likely produced positive pairs where one image contained a pleural line while the other did not, which we believe would make it difficult to learn representations for pleural line objects when pretraining with the StandardAug or AugUS-D pipelines. The performance of the frozen pretrained backbones was strong overall, considering the low capacity of the backbone and the small, narrow shape of pleural line objects. When fine-tuning end-to-end, SimCLR pretraining with AugUS-D resulted in the greatest test AP@50.

Backbone	Initial Weights	Pipeline	AP@50
Frozen	SimCLR	StandardAug	0.228
	SimCLR	AugUS-O	**0.255**
	SimCLR	AugUS-D	0.194
	Random	-	0.041
	ImageNet	-	0.127
Trainable	SimCLR	StandardAug	0.316
	SimCLR	AugUS-O	0.332
	SimCLR	AugUS-D	**0.351**
	Random	-	0.308
	ImageNet	-	0.310

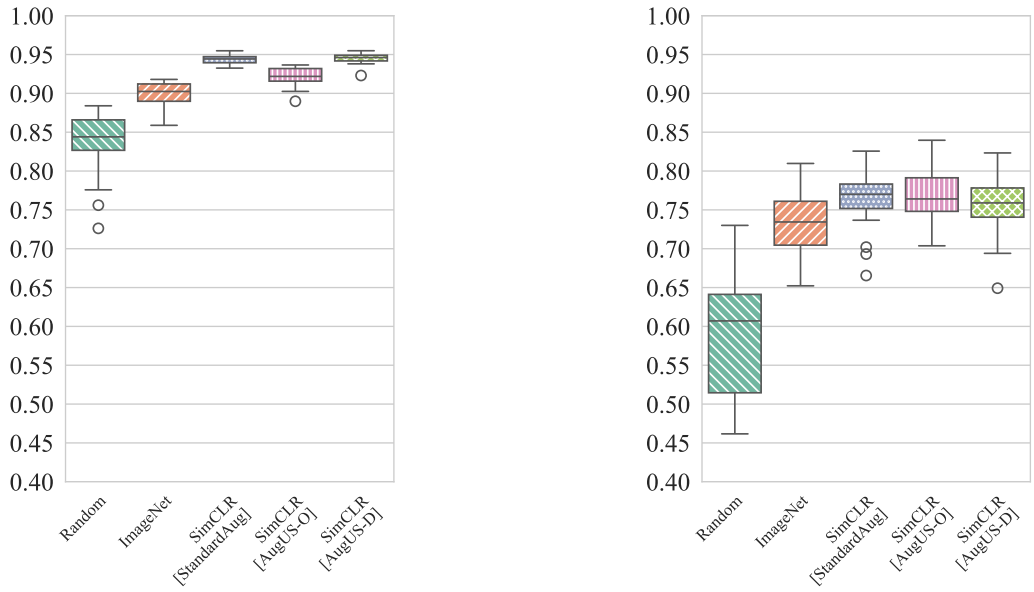
Table 6.10: LUSData local test set AP@50 for the PL task observed for SSD models whose backbones were pretrained using different data augmentation pipelines.

6.4.5 Label Efficiency Assessment

Experiments were conducted to test the robustness of pretrained models in settings where few labelled samples are available. The experiment was conducted only for the `AB` and `PE` classification tasks because there were enough unique videos and patients in the training set to create several disjoint subsets. Backbones were fine-tuned on 20 subsets of approximately 5% of the LUSData training set, split by patient, yielding 20 performance estimates for low-label settings. Splitting was conducted at the patient level to heighten the difficulty of the task and to limit dependence between subsets. Baseline estimates without SSL pretraining were obtained via initialization with random weights and with ImageNet-pretrained weights, resulting in five different performance conditions. Figure 6.6 displays boxplots for test AUC distributions under each condition. Friedman's Test indicated that there were significant differences among the median test AUC across conditions, for both the `AB` and `PE` tasks. Post-hoc Wilcoxon Sign-Rank Tests were then conducted for each pair of conditions, using the Bonferroni correction with a family-wise error rate of $\alpha = 0.05$. The median test AUCs of SimCLR-pretrained models were significantly greater than those initialized with random or ImageNet-pretrained weights for both the `AB` and `PE` tasks. All medians were significantly different for `AB`, except for the SimCLR-pretrained models using the StandardAug and AugUS-D pipelines, which achieved the greatest performance. Notably, these pipelines both utilize the crop and resize transformation. No significant differences were observed between any of the SimCLR-pretrained models for `PE`. Appendix K provides the test statistics for the above comparisons.

6.4.6 Impact of Semantics-Preserving Preprocessing

As outlined in Section 6.3.2, all ultrasound images were cropped to the smallest rectangle enclosing the FOV because the areas outside the FOV are bereft of information. Since pipelines containing the C&R transformation would be more likely to result in positive pairs that do not cover the FOV, it was hypothesized that cropping to the FOV as a preprocessing step would result in stronger pretrained backbones. To investigate the effect of this semantics-preserving preprocessing step, we pretrained backbones on LUSData using each data augmentation pipeline and evaluated them on the `AB`, `AB`, and `COVID` tasks. Table 6.11 compares the performance of each backbone with and without the preprocessing step. Performance on the `AB` task did not change. However, test AUC on both the `PE` and `COVID` tasks was consistently lower when the semantics-preserving preprocessing was not applied. Note that greatly less labelled data is available for `PE` and `COVID` than for `AB`. Based on these experiments, FOV cropping is a valuable semantics-preserving preprocessing step



(a) AB classification task

(b) PE classification task

Figure 6.6: Distribution of test AUC for classifiers trained on disjoint subsets of 5% of the patients in the training partition of the private dataset

for multiple LU classification tasks.

6.4.7 Impact of the Cropping Transformation in Object Classification Tasks

The leave-one-out analysis for transformations exhibited the striking finding that crop and resize (C&R) was the most effective transformation in the StandardAug pipeline for the two object classification tasks: AB and PE. Moreover, both pipelines containing C&R resulted in the greatest downstream test performance on AB and PE. Ordinarily, crops are taken at random locations in an image, with areas between 8% and 100% of the original image's area. Aggressive crops can create situations in which positive pairs do not contain the same objects of interest. Fig. 6.7 shows how C&R could produce positive pairs with different semantic content. Despite this, the results indicated that pipelines containing C&R led to improved performance for the object-centric AB and PE tasks. The exceptional influence of C&R warranted further investigation into its hyperparameters.

Task	Pipeline / Preprocessing	Linear classifier		Fine-tuned	
		✗	✓	✗	✓
AB	StandardAug	0.971	0.970	0.971	0.970
	AugUS-O	0.950	0.953	0.926	0.968
	AugUS-D	0.971	0.971	0.961	0.962
PE	StandardAug	0.873	0.893	0.869	0.882
	AugUS-O	0.846	0.865	0.522	0.557
	AugUS-D	0.867	0.897	0.864	0.884
COVID	StandardAug	0.742	0.784	0.724	0.753
	AugUS-O	0.793	0.836	0.805	0.836
	AugUS-D	0.585	0.713	0.737	0.744

Table 6.11: Test set AUC for SimCLR-pretrained models with (✓) and without (✗) semantics-preserving preprocessing. Results are reported for linear classifiers and fine-tuned models.

We investigated the impact of the minimum crop area, c, as a hyperparameter. Models were pretrained with the AugUS-D pipeline, using values for c in the range $[0.05, 0.9]$. Linear evaluation was conducted for the AB and PE tasks. As shown in Figure 6.8, smaller values of c yielded better performance, peaking at $c \approx 0.1$. The default assignment of $c = 0.08$ was already a reasonable choice for these two tasks.

Another concern with C&R is that it could result in crops covering the black background on images with a convex FOV. Despite the semantics-preserving preprocessing (described in Section 6.2), the top left and right corners of such images provide no information. To characterize the robustness of pretraining under these circumstances, we repeated the experiments sweeping over $c \in [0.05, 0.9]$ but first applied the probe type change transformation (i.e., U00) to every convex FOV. Thus, all inputs to the model were linear FOVs devoid of non-semantic background. A by-product of this transformation is that the near fields of convex images are horizontally stretched. As seen in Figure 6.8, this change resulted in a slight decrease in performance for both the AB and PE tasks. Evidently, the detriment of spatial distortion outweighed the benefit of guaranteeing that crops were positioned over semantic regions.

The C&R transform encourages pretrained representations to be invariant to scale. It is also believed that the C&R transform instills invariance between global and local views or between disjoint views of the same object type [29]. While the minimum area of

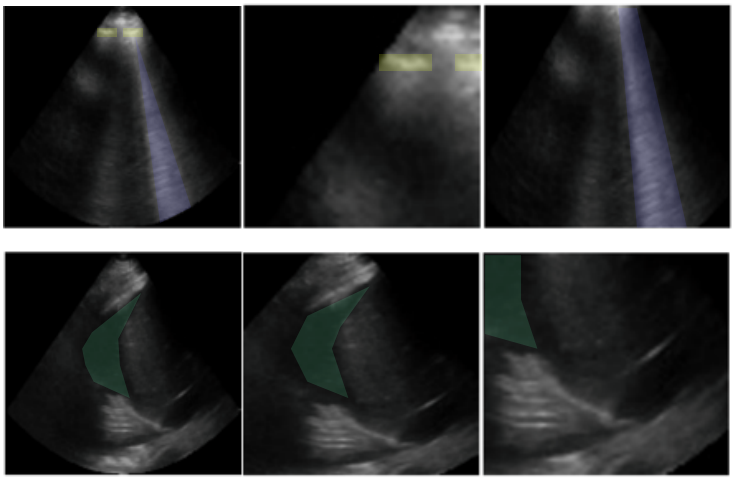
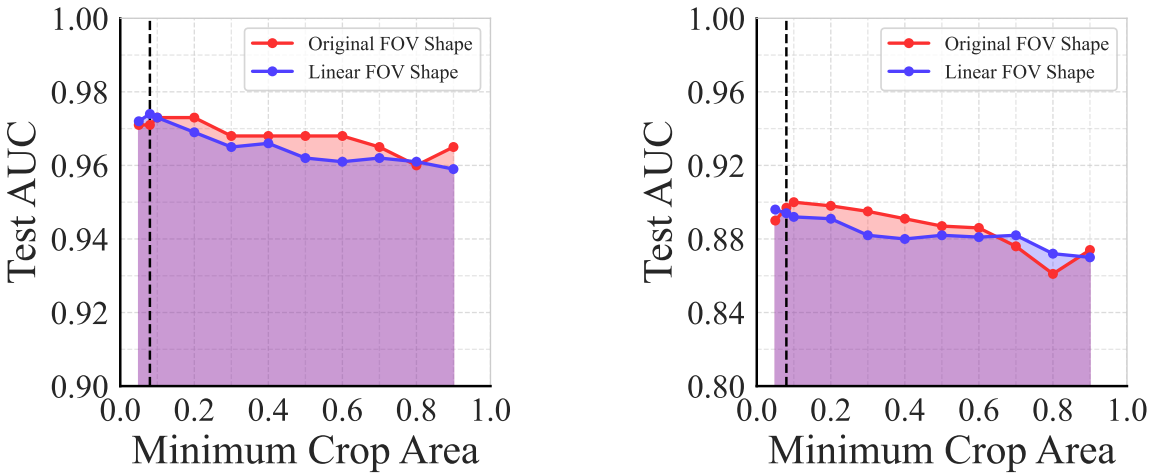


Figure 6.7: Examples of how the random crop and resize transformation (B00) can reduce semantic information. Original images are on the left, and two random crops of the image are on the right. **Top:** The original image contains a B-line (purple), which is visible in View 2 but not in View 1. The original image also contains instances of the pleural line (yellow) which are visible in View 1 but not in View 2. **Bottom:** The original image contains a pleural effusion (green), which is visible in View 1 but largely obscured in View 2.

the crop determines the magnitude of the scaling transformations, the aspect ratio range dictates the difference in distortion in both axes of the image. The default aspect ratio range is $[0.75, 1.33]$. We pretrained with the AugUS-D pipeline using a fixed aspect range of 1 and $c = 0.08$, which resulted in test AUC of 0.971 for `AB` and 0.881 for `PE` in the linear classification (LC) training setting. Compared to the regular AugUS-D that uses the default aspect ratio range (Table 6.7), `AB` test AUC remained unchanged, and `PE` test AUC decreased by 0.005.

Lastly, we conducted pretraining on LUSData using only the C&R transformation; that is, the data augmentation pipeline was *[B00]*. Recent work by Moutakanni *et al.* [115] suggests that, with sufficient quantities of training data, competitive performance in downstream computer vision tasks can be achieved using crop and resize as the sole transformation in joint embedding SSL. Linear evaluation of a feature extractor pretrained with only C&R yielded test AUC of 0.964 and 0.874 on `AB` and `PE`, respectively. Compared to the linear evaluations presented in Section 6.4.2, these metrics are greater than those achieved



(a) `AB` classification task

(b) `PE` classification task

Figure 6.8: Test set AUC for linear classifiers trained on the representations outputted by pretrained backbones. Each backbone was pretrained using AugUS-D with different values for the minimum crop area, c. Results are provided for models pretrained with the original ultrasound FOV, along with images transformed to linear FOV only. The dashed line indicates the default value of $c = 0.08$.

using AugUS-O, but less than those achieved with StandardAug or AugUS-D. It is evident that C&R is a powerful transformation for detecting local objects.

Overall, it is clear that aggressive C&R is beneficial for distinguishing between A-lines and B-lines and detecting pleural effusions on LU. Both are object-centric classification tasks. Even though some crops may not contain the object, the backbone would be exposed to several paired instances of transformed portions of objects during pretraining, potentially facilitating texture and shape recognition. Conversely, solving diagnostic tasks such as `COVID` requires a holistic assessment of the FOV, wherein the co-occurrence of objects is contributory to the overall impression.

6.5 Conclusion

In this chapter, we proposed and assessed data augmentation and preprocessing strategies for self-supervised learning applied to lung ultrasound tasks. A commonly employed baseline pipeline (StandardAug) was compared to a handcrafted semantics-preserving pipeline

(AugUS-O) and a hybrid pipeline (AugUS-D) composed from the first two. Evaluation on LU interpretation tasks revealed a dichotomy between the utility of the pipelines. Pipelines featuring the cropping transformation (StandardAug and AugUS-D) were most useful for object classification and detection tasks in LU. On the other hand, AugUS-O – designed to preserve semantics in LU – resulted in the greatest performance on a diagnostic task that required global context. Additionally, ultrasound field of view cropping was found to be a beneficial preprocessing step during pretraining for multiple LU classification tasks, regardless of the data augmentation strategy. Based on the results, machine learning practitioners should use semantics-preserving preprocessing during pretraining. Regarding data augmentation, semantics-preserving transformations should be considered for tasks requiring holistic interpretation of images, cropping-based transformations should be leveraged for object-centric downstream tasks.

Some limitations are acknowledged in this chapter. For example, SimCLR was the only SSL objective that was investigated, and all downstream tasks were confined to the lung. Moreover, some of the transformations introduced in this chapter constitute computationally expensive preprocessing steps, as they are applied with nonzero probability to each image.

Future work should apply this chapter's methods to assess the impact of data augmentation pipelines for ultrasound diagnostic tasks outside of the lung and for other SSL methods. Future studies could also compare data augmentation strategies for localization and segmentation downstream tasks in ultrasound.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

7.1.1 Primary Results

In this thesis, we examined the utility of self-supervised learning (SSL) for improving the performance of lung ultrasound (LU) machine learning classifiers. The work was motivated by two major obstacles that impeded the development of machine learning classifiers for LU interpretation: (1) the lack of labelled training data and (2) the paucity of experts capable of serving as labellers. SSL involves conducting supervised learning a neural network feature extractor with unlabelled data to solve a pretext task wherein targets are constructed from the unlabelled examples themselves. The evaluations spanned multiple core LU classification tasks on B-mode and M-mode, including view classification, A-line versus B-line classification, pleural effusion detection, lung sliding classification, and COVID-19 pneumonia classification. While SSL methods are being used in medical ultrasound, no studies have concentrated on integrating them into machine learning classifiers for basic findings in LU images.

To study the efficacy of self-supervised pretraining for LU tasks, contemporary joint embedding self-supervised learning methods were investigated. In Chapter 3, we observed that pretraining with contrastive and non-contrastive SSL boosted the performance of classifiers across multiple tasks in B-mode LU. We also observed that linear classifiers trained on the feature vectors outputted by pretrained feature extractors led to improved performance. The results motivate the development of pretrained feature extractors that can be used in a multi-task classification scheme for LU that avoids multiple invocations

of comparatively expensive neural network modules. In Chapter 4, it was observed that self-supervised pretraining was helpful for distinguishing between the absence and presence of the lung sliding sign on M-mode ultrasound (i.e., the `LS` task). To our knowledge, it was the first study that assessed SSL for any problem in M-mode ultrasound. An important finding across both chapters was that classifiers pretrained with SSL exhibited greater performance on test datasets originating from healthcare institutions that differed from the source of the training data. The improvement in generalizability was consistent with SSL studies concentrating on other medical imaging modalities.

After observing that pretraining with SSL established methods improved classifier performance on LU tasks, we concentrated on developing strategies that were tailored for ultrasound images. Informed by theoretical studies for SSL objectives, we posited novel positive pair sampling strategies. In Chapter 5, we presented a positive pair sampling method for ultrasound that leveraged the inherent structure of ultrasound as a temporal sequence of frames. In the context of contrastive learning, negative pairs were instances of images from different videos. In addition, we integrated sample weights into multiple SSL objectives, where sample weights were inversely proportional to the distance between intra-video pairs. While the experiments did not indicate that the sample weights had an impact, it was observed that the sampling strategy led to improved performance on multiple LU tasks.

Data augmentation is the predominant method for producing positive pairs in joint embedding SSL. In Chapter 6, we performed a comprehensive investigation into the efficacy of various data augmentation methods during SSL pretraining for core tasks in B-mode LU. A data augmentation pipeline for LU was proposed and compared to standard data augmentation methods. Leave-one-out analysis was also conducted to determine the most contributory transformations in each pipeline. Cropping was found to be the most impactful transformation with respect to downstream task performance. The results indicated that standard data augmentation transformations yielded greater results for object classification tasks in LU, while the ultrasound-tailored transformations led to greater results on the COVID-19 diagnostic classification task. Lastly, we demonstrated that field of view cropping during pretraining was a pivotal semantics-preserving preprocessing step. Chapter 2 explored semantics-preserving data augmentation for M-mode, finding that they provided greater benefit for the downstream lung sliding classification task when using randomly initialized weights prior to pretraining. However, standard data augmentation provided greater gains when initializing with ImageNet-pretrained weights prior to self-supervision.

7.1.2 Recommendations

Based on the results of the investigations, we provide the following overarching recommendations for machine learning practitioners working with LU data:

R1. When unlabelled videos are available, pretraining using SSL methods is a good strategy to try before launching expensive data labelling campaigns. Pretraining with large datasets of unlabelled images and then supervised fine-tuning on small, labelled training sets can be as effective as supervised learning using larger labelled training sets.

See Sections 3.4 and 4.4; see also [155, 156].

R2. Pretraining using the SimCLR contrastive learning method [29] tends to provide the greatest improvement in downstream performance on basic LU downstream tasks.

See Section 5.4.2; see also [157].

R3. Distributional shifts are common in ultrasound images across different healthcare institutions, due to factors such as different ultrasound manufacturer use, patient demographics, and operator techniques. In cases where models are to be deployed in external healthcare institutions, SSL pretraining may increase generalizability.

See Sections 3.4, 4.4.4, and 6.4.2; see also [154–156].

R4. Consistent with observations across other medical imaging modalities [9], initializing feature extractors with ImageNet-pretrained weights prior to SSL pretraining produces better results than starting with random weights.

See Sections 4.4.2 and 5.5; see also [156, 157].

R5. Intra-video positive pairs is a viable positive pair sampling strategy. However, hyperparameter searches should be conducted to yield maximum benefit.

See Sections 5.4 and 5.5; see also [157].

R6. Cropping ultrasound images to their field of view is a beneficial semantics-preserving preprocessing step during SSL.

See Section 6.4.6; see also [154].

R7. Semantics-preserving ultrasound-tailored data augmentation may be more suited to diagnostic LU tasks, while standard cropping-based data augmentation would be a

good first choice for local object classification tasks. Greater breadth in LU dataset annotations would improve the clarity of this recommendation.

See Sections 6.4.2 and 6.4.3; see also [154].

R8. Random cropping is a highly effective data augmentation transformation for downstream LU object classification tasks. Setting a small minimum crop area of approximately 10% of image area produced the greatest results for B-mode LU classification tasks.

See Section 6.4.7; see also [154].

7.2 Directions for Future Work

The results of this thesis spawn several avenues for future investigation. SSL is a burgeoning area of research in machine learning; as such, there are several ideas to explore that could be integrated into machine learning systems designed for POCUS or that could be extended to suit the modality. In this section, we present some avenues for future study.

7.2.1 Masked Image Modelling for Lung Ultrasound

First, there is a growing interest in the masked image modelling (MIM) paradigm of SSL. Succinctly, the pretext task consists of training an autoencoder to predict the content of redacted portions of images. The backbones correspond to the encoder component of the autoencoder, and they are typically vision transformers. The autoencoders are trained to predict masked tokens that correspond to redacted patches in images. These methods have demonstrated competitive performance with joint embedding SSL methods [66, 168]. MIM has been recently applied in breast ultrasound [98, 130, 170] and thyroid ultrasound [80]. The efficacy of these methods should be explored for ultrasound of the lung as well. Since many masking approaches require transformer encoders and decoders, they may require more pretraining data to achieve competitive performance.

Masked Image Modelling Versus Joint Embedding Methods

To study the efficacy of MIM for LU, future work could compare an assortment of predominant methods for classification tasks in LU, guided by the following questions:

1. How do MIM methods compare with joint embedding methods with respect to downstream performance on a variety of LU tasks?

2. Recall that MIM methods typically use vision transformer backbones. Accordingly, do MIM methods require more pretraining data to perform comparably to CNNs pretrained with joint embedding methods to achieve comparable performance? A future investigation could assess the volume of pretraining data when comparing test set metrics for fine-tuned backbones produced using each class of methods.

The outcome of these investigations would be a set of recommendations for whether to use joint embedding or MIM approaches under various circumstances for pretraining data availability.

Time Masks Versus Spatial Masks for M-Mode

None of the aforementioned applications of MIM approaches have been applied to M-mode ultrasound. As discussed in this thesis, M-mode is an important modality in LU that is useful in the diagnostic workup of patients with suspected pneumothorax. Most MIM approaches use a similar masking strategy — images are randomly divided into square patches and a random selection of patches are redacted and the remaining patches are inputted into the encoder. Recent work by Xu *et al.* [170] studied alternative masking strategies for B-mode ultrasound. In addition to randomly sampling patch locations, they explored confining masked patches to randomly selected columns or rows of patches across the image. No evidence clearly favoured any one strategy. Unlike B-mode, M-mode ultrasound has one spatial and one time dimension. As such, a future investigation could explore whether column- or row-based masking strategies are useful for the lung sliding (`LS`) downstream task. Analogous to missing frame prediction in videos, we hypothesize that column-based masking would yield strong representations for `LS`, as the backbone would be required to learn how artefacts evolve over time. The study could also examine the effect of the fraction of masked patches in the image (i.e., the mask ratio) on the downstream performance of classifiers.

Lastly, since the seashore and barcode signs are fine-grained textures, generative SSL methods may be generally difficult for M-mode LU due to the reconstructive pretraining loss functions. One avenue could be to explore the effect of the patch size on `LS` performance, as a smaller patch size could facilitate the learning of fine-grained representations. Another option would be to employ a reconstruction loss function in the frequency domain, in addition to or in place of reconstruction loss in the spatial domain. Learning to

produce reconstructions that share similar spectra to the original M-mode images could be advantageous for learning representations that differ between the seashore and barcode signs, since the seashore sign's sandy appearance corresponds to high-frequency regions.

Based on the discussion above, we identify the following research questions:

1. Do MIM methods improve downstream LS performance on LU M-mode images? If so, which masking strategy yields the greatest performance: standard patch masking, row masking, or column masking?
2. Does the patch size during pretraining affect downstream LS performance?
3. Does replacing or augmenting the standard pixel reconstruction loss with a reconstruction loss in the frequency domain have an impact on downstream LS performance?

7.2.2 Video-Level Self-Supervised Learning

Although the LU interpretation tasks studied in this thesis can be conducted by examining still frames from videos, the temporal context of whole video interpretation may further improve performance. Many image-based SSL methods and ideas have been adapted for video inputs. Video-level adaptations of MIM methods with large transformers have recently demonstrated strong performance on action recognition tasks [135]. These methods typically operate on fixed-duration subsections of videos. Enormous volumes of pretraining data would likely be required to learn strong feature extractors, as the capacity of model architectures that directly ingest video inputs is generally much greater than those that operate on images.

While Chapters 3 and 5 exploited the video structure of B-mode ultrasound to create positive pairs, we did not explore video-level SSL in this thesis. Although some of these methods have been explored for some tasks in ultrasound [28, 185], it remains unclear whether video-level representations are superior to image-level representations. There would be some beneficial aspects of learning video-level representations for LU. Some patterns and artefacts disappear and reappear throughout the duration of videos, which presents logistical challenges when working with image-level classifiers. B-lines are an example in LU of a transient artefact. Prior work on LU video classification addressed these challenges by training image classifiers using only image from videos that exhibit the same findings throughout and devising clinically informed algorithms that produce a video-level prediction from a series of image-level prediction confidences [6, 26, 152]. For

example, if there exists any contiguous set of n frames in a video for which the prediction confidence exceeds a classification threshold for B-lines, then the video-level prediction is B-lines [6]. Note that all labelled datasets in this thesis possessed video-level labels that applied to each constituent image.

Future work should explore video-based SSL in greater detail for POCUS interpretation tasks. Specifically, a comprehensive study of video-level SSL compared to image-level counterparts could be conducted for a variety of tasks in POCUS to ascertain the following:

1. What scale of unlabelled data are required to obtain clinically acceptable metrics? How does it compare to the scale required for pretrained image classifiers whose prediction confidences are combined to a single video decision using a clinically informed algorithm?

2. What is the optimal duration and downsampled frame rate for video inputs? Do the optimal values differ for various downstream tasks?

3. How does the inference efficiency of video backbones compare to batched inference with image backbones? The experiments should compare metrics such as elapsed real time, floating-point operations, parameter count, and memory usage. In edge deployments for ultrasound, lightweight architectures are favoured to ensure local inference.

Note that while these investigations can be conducted on LU datasets, they should be applied to several ultrasound examinations outside of the lung.

7.2.3 Reusable Backbones for Efficient Multi-Task Inference

A key design challenge in building utilitarian systems for automated LU interpretation is maintaining one of POCUS's main strengths — portability. A fundamental differentiator of POCUS is the fact that it can be conducted anywhere, including austere environments. Settings where LU is useful may have unreliable or even non-existent internet connection, precluding cloud-based interpretation services. As a result, offline inference on edge devices is highly desirable.

In Chapter 3, we explored the efficacy of a LU interpretation workflow in which a shared backbone produced features that were passed to multiple lightweight classifiers. The forward pass through the backbone was, by far, the most expensive component of the design. Adding lightweight classifiers such as shallow MLPs for additional tasks does not

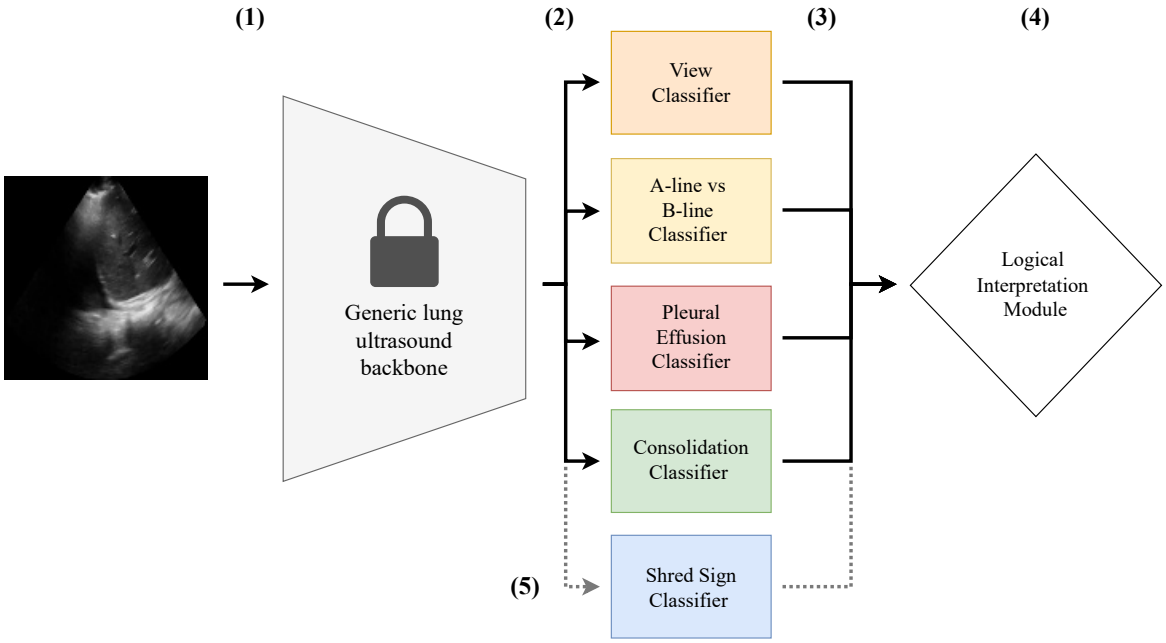


Figure 7.1: A depiction of a multi-task LU interpretation system centred around one pretrained backbone. **(1)** A LU image is passed to a backbone pretrained with SSL. The backbone's weights are frozen after pretraining and are not used for fine-tuning. **(2)** The backbone outputs a feature vector containing semantic information. **(3)** Multiple compact classifiers produce predictions for various LU tasks based on the same feature vector. **(4)** Clinical algorithms use the task-specific predictions to output relevant interpretation for the image. **(5)** A classifier for a new task can be added to the system without repeating self-supervised pretraining.

materially increase the resources or time required for inference. Chapter 6 also provided evidence in favour of backbone versatility, as some linear classifiers trained using feature vectors from frozen backbones outperformed fine-tuned classifiers.

Future work could delve into the optimization of this multi-task inference architecture. As sketched out in Figure 7.1, the design would centre around a single backbone that outputs feature vectors containing enough semantic information to effectively train classifiers for multiple findings in LU. Analogous to how pluripotent stem cells in the body can be induced to develop into several cell types, a generic frozen backbone's features could be used to develop classifiers for numerous tasks. The investigation could quantify the complexity of the lightweight classifiers required to match full fine-tuning. Research questions

that could be addressed include:

1. What are the number of layers and number of nodes per layer of MLPs that ingest feature vectors from pretrained backbones that achieve clinically acceptable performance for a variety of downstream tasks?

2. Task-specific fine-tuning of a limited number of final layers in the backbone are an alternative avenue to adding MLPs. For core architectures (e.g., MobileNetV3Small), how many blocks must be fine-tuned to achieve clinically acceptable performance on a set of downstream tasks?

3. What effect do architectural attributes and hyperparameters of the backbone have on multi-task downstream task performance? For example, do highly parameterized backbones extract features that are more versatile for multiple tasks?

To conduct a standalone evaluation of ensuing designs, a classifier could be trained for a held-out task that was not used to inform design choices of the multi-task design.

Lastly, the use of SSL for multi-task inference can readily be applied to other domains. Similar investigations could be conducted in domains unrelated to ultrasound where inference on edge devices is valuable, such as speech analysis or photographic image classification.

7.2.4 Does Self-Supervised Learning Improve Generalizability?

Finally, yet importantly, the use of SSL as a means to improve the generalizability of LU machine learning systems should be explored in greater detail. Out-of-distribution performance is a wicked problem that affects medical imaging analysis software modules that are developed using machine learning. The problem is pronounced in ultrasound imaging, as there are several sources of distributional shift across healthcare institutions, such as patient demographics, probe manufacturers, and operator training/practice. The external test set in Chapter 6 was an illustrative example of differences in device manufacturer distributions across clinical sites (see Table 6.3). When developing a product that examines patterns in medical images, external validation is paramount to meet regulatory requirements. For example, the Food and Drug Administration recommends that test data sites must be independent from training data sites [151].

Across this thesis, we observed that LU classifiers pretrained with SSL tended to achieve greater performance on test sets from external institutions than those that were only trained

with supervised learning, even when initialized with ImageNet-pretrained weights. As outlined in Section 2.4.3, self-supervised pretrained DNNs have attained greater performance metrics on external test sets than baselines without self-supervision across a variety of medical imaging modalities [8, 9, 150, 195].

Subsequent studies should focus on characterizing the phenomenon of improved out-of-distribution performance and testing its consistency across different pattern recognition tasks in POCUS. Specifically, the following questions should be addressed:

1. Do classifiers pretrained with SSL consistently exhibit improved performance on external test sets acquired from other institutions across multiple POCUS examinations (e.g., abdominal, echocardiography)?

2. The external validation experiments in this thesis were only concerned with classification tasks. Do object detection or segmentation DNNs benefit from self-supervised pretraining with respect to out-of-distribution performance?

3. Is generalizability to external test sets a result of exposing backbones to a greater variety of examples during pretraining? For example, suppose that the pretraining dataset contains unlabelled videos acquired using a diverse set of probe manufacturers, M_{pre}. Does external test set performance improve if its set of manufacturers M_{ext} is a subset of M_{pre}? Does this hold even if the set of manufacturers represented in the training set, M_{tr}, is different (i.e., $M_{\text{tr}} \cap M_{\text{ext}} = \emptyset$)?

4. Is there a theoretical explanation that is consistent with the observation that self-supervised pretraining yields improves performance on out-of-distribution datasets?

Understanding why SSL supports generalizability to external data distributions could create invaluable knowledge that would help machine learning developers produce safer software-based medical devices that operate under a greater variety of conditions and populations. Moreover, it could inform future regulatory requirements for software-based medical devices that contain DNN modules.

7.3 Closing Words

This thesis has investigated the use of self-supervised learning and unlabelled data in the development of lung ultrasound machine learning systems. It has demonstrated that self-supervised pretraining is a beneficial practice for building effective and generalizable

machine learning classifiers. In diagnostic imaging, every improvement matters, as better outcomes can be delivered for both patients and providers. We hope that the results and recommendations presented in this thesis lead to the design of strong systems for portable, safe, and affordable respiratory diagnostics.

References

[1] Walid Al-Dhabyani, Mohammed Gomaa, Hussien Khaled, and Aly Fahmy. Dataset of breast ultrasound images. *Data in brief*, 28:104863, 2020. Publisher: Elsevier.

[2] Khaled Alrajhi, Michael Y Woo, and Christian Vaillancourt. Test characteristics of ultrasonography for the detection of pneumothorax: A systematic review and meta-analysis. *Chest*, 141(3):703–708, 2012. Publisher: Elsevier.

[3] Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Lies Hadjadj, Emilie Devijver, and Yury Maximov. Self-training: A survey. *Neurocomputing*, 616:128904, 2025.

[4] D. Anand, P. Annangi, and P. Sudhakar. Benchmarking self-supervised representation learning from a million cardiac ultrasound images. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, volume 2022-July, pages 529–532. Institute of Electrical and Electronics Engineers Inc., 2022.

[5] Robert Arntfield, Blake VanBerlo, Thamer Alaifan, Nathan Phelps, Matthew White, Rushil Chaudhary, Jordan Ho, and Derek Wu. Development of a convolutional neural network to differentiate among the etiology of similar appearing pathological B lines on lung ultrasound: a deep learning study. *BMJ open*, 11(3):e045120, 2021. Publisher: British Medical Journal Publishing Group.

[6] Robert Arntfield, Derek Wu, Jared Tschirhart, Blake VanBerlo, Alex Ford, Jordan Ho, Joseph McCauley, Benjamin Wu, Jason Deglint, Rushil Chaudhary, and others. Automation of lung ultrasound interpretation via deep learning for the classification of normal versus abnormal lung parenchyma: A multicenter study. *Diagnostics*, 11(11):2049, 2021. Publisher: MDPI.

[7] S. Atito, S.M. Anwar, M. Awais, and J. Kittler. SB-SSL: Slice-based self-supervised transformers for knee abnormality classification from MRI. *Lecture Notes in Com-*

puter Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 13559 LNCS:86–95, 2022.

[8] Shekoofeh Azizi, Laura Culp, Jan Freyberg, Basil Mustafa, Sebastien Baur, Simon Kornblith, Ting Chen, Nenad Tomasev, Jovana Mitrović, Patricia Strachan, et al. Robust and data-efficient generalization of self-supervised machine learning for diagnostic imaging. *Nature Biomedical Engineering*, 7(6):756–779, 2023.

[9] Shekoofeh Azizi, Basil Mustafa, Fiona Ryan, Zachary Beaver, Jan Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, Ting Chen, and others. Big self-supervised models advance medical image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3478–3488, 2021.

[10] Cagla Deniz Bahadir, Mohamed Omar, Jacob Rosenthal, Luigi Marchionni, Benjamin Liechty, David J. Pisapia, and Mert R. Sabuncu. Artificial intelligence applications in histopathology. *Nature Reviews Electrical Engineering*, February 2024.

[11] Randall Balestriero and Yann LeCun. Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 26671–26685. Curran Associates, Inc., 2022.

[12] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-Invariance-Covariance Regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

[13] Soumen Basu, Somanshu Singla, Mayank Gupta, Pratyaksha Rana, Pankaj Gupta, and Chetan Arora. Unsupervised contrastive learning of image representations from ultrasound videos with hard negative mining. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part IV*, pages 423–433. Springer, 2022.

[14] Joseph Geo Benjamin, Mothilal Asokan, Amna Alhosani, Hussain Alasmawi, Werner Gerhard Diehl, Leanne Bricker, Karthik Nandakumar, and Mohammad Yaqub. Leveraging self-supervised learning for fetal cardiac planes classification using ultrasound scan videos. In *International Workshop on Advances in Simplifying Medical Ultrasound*, pages 68–78. Springer, 2023.

[15] Lucien Birgé and Pascal Massart. From model selection to adaptive estimation. In David Pollard, Erik Torgersen, and Grace L. Yang, editors, *Festschrift for Lucien Le Cam: Research Papers in Probability and Statistics*, pages 55–87. Springer, New York, NY, 1997.

[16] Ivana Blazic, Chiara Cogliati, Nicola Flor, Guy Frija, Michael Kawooya, Michele Umbrello, Sam Ali, Marie-Laure Baranne, Young-Jae Cho, Richard Pitcher, and others. The use of lung ultrasound in COVID-19. *ERJ Open Research*, 9(1), 2023. Publisher: European Respiratory Society.

[17] Florian Bordes, Randall Balestriero, Quentin Garrido, Adrien Bardes, and Pascal Vincent. Guillotine Regularization: Why removing layers is needed to improve generalization in self-supervised learning. *Transactions on Machine Learning Research*, 2023.

[18] Florian Bordes, Randall Balestriero, and Pascal Vincent. Towards democratizing joint-embedding self-supervised learning. *arXiv preprint arXiv:2303.01986*, 2023.

[19] Jannis Born, Gabriel Brändle, Manuel Cossio, Marion Disdier, Julie Goulet, Jérémie Roulin, and Nina Wiedemann. POCOVID-Net: Automatic detection of COVID-19 from a new lung ultrasound imaging dataset (POCUS). *arXiv preprint arXiv:2004.12084*, 2020.

[20] Morten Thingemann Bøtker, Lars Jacobsen, Søren Steemann Rudolph, and Lars Knudsen. The role of point of care ultrasound in prehospital critical care: A systematic review. *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, 26:1–14, 2018.

[21] Anna K Brady, Carleen R Spitzer, Diana Kelm, Shari B Brosnahan, Mani Latifi, and Kristin M Burkart. Pulmonary critical care fellows' use of and self-reported barriers to learning bedside ultrasound during training: results of a national survey. *Chest*, 160(1):231–237, 2021. Publisher: Elsevier.

[22] Butterfly Network. COVID-19 ultrasound gallery. https://drive.google.com/file/d/1zefZInevopumI-VdX6r7Bj-6pj_WILrr/view?usp=sharing, 2020. Mirror hosted via the 983632847/USCL GitHub repository; accessed October 12, 2023.

[23] Vivien Cabannes, Bobak Kiani, Randall Balestriero, Yann LeCun, and Alberto Bietti. The SSL interplay: Augmentations, inductive bias, and generalization. In *International Conference on Machine Learning*, pages 3252–3298. PMLR, 2023.

[24] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[25] F. Chang, C. Wu, Y. Wang, Y. Zhang, X. Chen, and Q. Tian. Boundary-enhanced self-supervised learning for brain structure segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13431 LNCS:14–23, 2022.

[26] Rushil Chaudhary, Jordan Ho, Delaney Smith, Saad Hossain, Jaswin Hargun, Blake VanBerlo, Niall Murphy, Ross Prager, Kiran Rikhraj, Jared Tschirhart, et al. Diagnostic accuracy of an automated classifier for the detection of pleural effusions in patients undergoing lung ultrasound. *The American Journal of Emergency Medicine*, 2025.

[27] L. Chen, P. Bentley, K. Mori, K. Misawa, M. Fujiwara, and D. Rueckert. Self-supervised learning for medical image analysis using image context restoration. *Medical Image Analysis*, 58, 2019.

[28] Li Chen, Jonathan Rubin, Jiahong Ouyang, Naveen Balaraju, Shubham Patil, Courosh Mehanian, Sourabh Kulhare, Rachel Millin, Kenton W Gregory, Cynthia R Gregory, and others. Contrastive self-supervised learning for spatio-temporal analysis of lung ultrasound videos. In *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2023.

[29] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[30] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[31] Xinlei Chen and Kaiming He. Exploring simple Siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021.

[32] Yixiong Chen, Chunhui Zhang, Chris HQ Ding, and Li Liu. Generating and weighting semantically consistent sample pairs for ultrasound contrastive learning. *IEEE Transactions on Medical Imaging*, 2022.

[33] Yixiong Chen, Chunhui Zhang, Li Liu, Cheng Feng, Changfeng Dong, Yongfang Luo, and Xiang Wan. USCL: Pretraining deep ultrasound image diagnosis model through video contrastive representation learning. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part VIII 24*, pages 627–637. Springer, 2021.

[34] Davide Chiumello, Michele Umbrello, Giuseppe Francesco Sferrazza Papa, Alessio Angileri, Martina Gurgitano, Paolo Formenti, Silvia Coppola, Sara Froio, Antonio Cammaroto, and Gianpaolo Carrafiello. Global and regional diagnostic accuracy of lung ultrasound compared to CT in patients with acute respiratory distress syndrome. *Critical care medicine*, 47(11):1599–1606, 2019. Publisher: LWW.

[35] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009.

[36] N Desai and T Harris. Extended focused assessment with sonography in trauma. *BJA education*, 18(2):57–62, 2018.

[37] Fatemeh Taheri Dezaki, Christina Luong, Tom Ginsberg, Robert Rohling, Ken Gin, Purang Abolmaesumi, and Teresa Tsang. Echo-SyncNet: Self-supervised cardiac view synchronization in echocardiography. *IEEE Transactions on Medical Imaging*, 40(8):2092–2104, August 2021.

[38] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1422–1430, Los Alamitos, CA, USA, December 2015. IEEE Computer Society. ISSN: 2380-7504.

[39] Wenju Du, Nini Rao, Dingyun Liu, Hongxiu Jiang, Chengsi Luo, Zhengwen Li, Tao Gan, and Bing Zeng. Review on the applications of deep learning in the analysis of gastrointestinal Endoscopy images. *IEEE Access*, 7:142053–142069, 2019.

[40] Christophe Dubecq, Olivier Dubourg, Gabriel Morand, Romain Montagnon, Stéphane Travers, and Pierre Mahe. Point-of-care ultrasound for treatment and triage in austere military environments. *Journal of Trauma and Acute Care Surgery*, 91(2S):S124–S129, 2021.

[41] B. Dufumier, P. Gori, J. Victor, A. Grigis, M. Wessa, P. Brambilla, P. Favre, M. Polosan, C. McDonald, C.M. Piguet, M. Phillips, L. Eyler, E. Duchesnay, and

the Alzheimer's Disease Neuroimaging Initiative. Contrastive learning with continuous proxy meta-data for 3D MRI classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12902 LNCS:58–68, 2021.

[42] Nabeel Durrani, Damjan Vukovic, Jeroen van der Burgt, Maria Antico, Ruud JG van Sloun, David Canty, Marian Steffens, Andrew Wang, Alistair Royse, Colin Royse, and others. Automatic deep learning-based consolidation/collapse classification in lung ultrasound images for COVID-19 induced pneumonia. *Scientific Reports*, 12(1):17581, 2022. Publisher: Nature Publishing Group UK London.

[43] Ashkan Ebadi, Pengcheng Xi, Alexander MacLean, Adrian Florea, Stéphane Tremblay, Sonny Kohli, and Alexander Wong. COVIDx-US: An open-access benchmark dataset of ultrasound imaging data for AI-driven COVID-19 analytics. *Front. Biosci. (Landmark Ed.)*, 27(7):198, June 2022. Publisher: IMR Press.

[44] Aciel Eshky, Manuel Sam Ribeiro, Joanne Cleland, Korin Richmond, Zoe Roxburgh, James Scobbie, and Alan Wrench. UltraSuite: A repository of ultrasound and acoustic data from child speech therapy sessions. *arXiv preprint arXiv:1907.00835*, 2019.

[45] R. Feng, Z. Zhou, M.B. Gotway, and J. Liang. Parts2Whole: Self-supervised contrastive learning via reconstruction. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12444 LNCS:85–95, 2020.

[46] Alvaro Fernandez-Quilez, Trygve Eftestøl, Svein Reidar Kjosavik, Morten Goodwin, and Ketil Oppedal. Contrasting axial T2W MRI for prostate cancer triage: A self-supervised learning approach. In *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2022.

[47] Hans Clausdorff Fiedler, Ross Prager, Delaney Smith, Derek Wu, Chintan Dave, Jared Tschirhart, Ben Wu, Blake Van Berlo, Richard Malthaner, and Robert Arntfield. Automated real-time detection of lung sliding using artificial intelligence: A prospective diagnostic accuracy study. *Chest*, 166(2):362–370, 2024.

[48] Milton Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The annals of mathematical statistics*, 11(1):86–92, 1940. Publisher: JSTOR.

[49] Guanghui FU, Ruiqian Wang, Jianqiang Li, Maria Vakalopoulou, and Vicky Kalogeiton. Me-NDT: Neural-backed decision tree for visual explainability of deep medical models. In *Medical Imaging with Deep Learning*, 2021.

[50] Lulu Gai, Wei Chen, Rui Gao, Yan-Wei Chen, and Xu Qiao. Using vision transformers in 3-D medical image classifications. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 696–700, October 2022.

[51] Luna Gargani. Lung ultrasound: A new tool for the cardiologist. *Cardiovascular ultrasound*, 9:1–9, 2011.

[52] Quentin Garrido, Yubei Chen, Adrien Bardes, Laurent Najman, and Yann LeCun. On the duality between contrastive and non-contrastive self-supervised learning. *arXiv preprint arXiv:2206.02574*, 2022.

[53] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018.

[54] Amy Sarah Ginsburg, Zylee Liddy, Parsa T Khazaneh, Susanne May, and Farhan Pervaiz. A survey of barriers and facilitators to ultrasound use in low-and middle-income countries. *Scientific Reports*, 13(1):3322, 2023. Publisher: Nature Publishing Group UK London.

[55] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. Publisher: ACM New York, NY, USA.

[56] Bhawna Goyal, Sunil Agrawal, and BS Sohi. Noise issues prevailing in various types of medical images. *Biomedical & Pharmacology Journal*, 11(3):1227, 2018.

[57] Cynthia A Gravel, Michael C Monuteaux, Jason A Levy, Andrew F Miller, Rebecca L Vieira, and Richard G Bachur. Interrater reliability of pediatric point-of-care lung ultrasound findings. *The American Journal of Emergency Medicine*, 38(1):1–6, 2020.

[58] Yonatan Y Greenstein and Keith Guevarra. Point-of-care ultrasound in the intensive care unit: Applications, limitations, and the evolution of clinical practice. *Clinics in Chest Medicine*, 43(3):373–384, 2022. Publisher: Elsevier.

[59] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, and others. Bootstrap Your own Latent - a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

[60] Yang Guo, Yuanbo He, Shuai Li, Ting Shu, and Luying Gao. Global correlation and local geometric information coupled channel contrast learning for thyroid nodule risk stratification. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 868–875, December 2021.

[61] Manoj Gupta, Heena Taneja, and Laxmi Chand. Performance enhancement and analysis of filters in ultrasound image denoising. *Procedia Computer Science*, 132:643–652, 2018.

[62] Fatemeh Haghighi, Mohammad Reza Hosseinzadeh Taher, Michael B. Gotway, and Jianming Liang. DiRA: Discriminative, restorative, and adversarial learning for self-supervised medical image analysis. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20792–20802, June 2022.

[63] M Kennedy Hall, Jane Hall, Cary P Gross, Nir J Harish, Rachel Liu, Sean Maroongroge, Christopher L Moore, Christopher C Raio, and R Andrew Taylor. Use of point-of-care ultrasound in the emergency department: Insights from the 2012 Medicare national payment data set. *Journal of Ultrasound in Medicine*, 35(11):2467–2474, 2016. Publisher: Wiley Online Library.

[64] Louise Hansell, Maree Milross, Anthony Delaney, David H Tian, Arvind Rajamani, and George Ntoumenopoulos. Barriers and facilitators to achieving competence in lung ultrasound: a survey of physiotherapists following a lung ultrasound training course. *Australian Critical Care*, 36(4):573–578, 2023.

[65] Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011, 2021.

[66] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[67] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the*

IEEE/CVF conference on computer vision and pattern recognition, pages 9729–9738, 2020.

[68] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[69] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

[70] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979. Publisher: JSTOR.

[71] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, and others. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[72] Szu-Yen Hu, Shuhang Wang, Wei-Hung Weng, JingChao Wang, XiaoHong Wang, Arinc Ozturk, Quan Li, Viksit Kumar, and Anthony E Samir. Self-supervised pretraining with DICOM metadata in ultrasound imaging. In *Machine Learning for Healthcare Conference*, pages 732–749. PMLR, 2020.

[73] H. Huang, R. Wu, Y. Li, and C. Peng. Self-supervised transfer learning based on domain adaptation for benign-malignant lung nodule classification on thoracic CT. *IEEE Journal of Biomedical and Health Informatics*, 26(8):3860–3871, 2022.

[74] J. Huang, H. Li, G. Li, and X. Wan. Attentive symmetric autoencoder for brain MRI segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13435 LNCS:203–213, 2022.

[75] Kristin Ireland, Noha Daher, Michael Terry, David López, Paul Casillas, Laren D Tan, and Abdullah Alismail. Perception, challenges, and barriers of point-of-care lung ultrasound among respiratory therapist in the United States. *CHEST Pulmonary*, 2(1):100029, 2024.

[76] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, and

others. CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019. Issue: 01.

[77] Ananya Jana, Hui Qu, Carlos D. Minacapelli, Carolyn Catalano, Vinod Rustgi, and Dimitris Metaxas. Liver fibrosis and NAS scoring from CT images using self-supervised learning and texture encoding. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1553–1557, April 2021.

[78] Miroslav Jaščur, Marek Bundzel, Marek Malík, Anton Dzian, Norbert Ferenčík, and František Babič. Detecting the absence of lung sliding in lung ultrasounds using deep learning. *Applied Sciences*, 11(15):6976, 2021. Publisher: MDPI.

[79] J. Jiao, R. Droste, L. Drukker, A.T. Papageorghiou, and J.A. Noble. Self-supervised representation learning for ultrasound video. In *Proceedings - International Symposium on Biomedical Imaging*, volume 2020-April, pages 1847–1850. IEEE Computer Society, 2020.

[80] Qingbo Kang, Jun Gao, Kang Li, and Qicheng Lao. Deblurring masked autoencoder is better recipe for ultrasound image recognition. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 352–362. Springer, 2023.

[81] Onur Karaoğlu, Hasan Şakir Bilge, and İhsan Uluer. Removal of speckle noises from ultrasound images using five different deep learning networks. *Engineering Science and Technology, an International Journal*, 29:101030, 2022.

[82] S. Kayal, S. Chen, and M. de Bruijne. Region-of-interest guided supervoxel inpainting for self-supervision. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12261 LNCS:500–509, 2020.

[83] Brendan S Kelly, Conor Judge, Stephanie M Bollard, Simon M Clifford, Gerard M Healy, Awsam Aziz, Prateek Mathur, Shah Islam, Kristen W Yeom, Aonghus Lawlor, and others. Radiology artificial intelligence: A systematic review and evaluation of methods (RAISE). *European radiology*, 32(11):7998–8007, 2022. Publisher: Springer.

[84] Ross Kessler, Jeffrey R Stowell, Jody A Vogel, Michael M Liao, and John L Kendall. Effect of interventional program on the utilization of PACS in point-of-care ultrasound. *Journal of digital imaging*, 29:701–705, 2016. Publisher: Springer.

[85] Kyungsang Kim, Fabiola Macruz, Dufan Wu, Christopher Bridge, Suzannah McKinney, Ahad Alhassan Al Saud, Elshaimaa Sharaf, Adam Pely, Paul Danset, Tom Duffy, and others. Point-of-care AI-assisted stepwise ultrasound pneumothorax diagnosis. *Physics in Medicine & Biology*, 68(20):205013, 2023. Publisher: IOP Publishing.

[86] X. Kuang, J.P.Y. Cheung, X. Ding, and T. Zhang. SpineGEM: A hybrid-supervised model generation strategy enabling accurate spine disease classification with a small training dataset. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12902 LNCS:145–154, 2021.

[87] Sarah Leclerc, Erik Smistad, Joao Pedrosa, Andreas Østvik, Frederic Cervenansky, Florian Espinosa, Torvald Espeland, Erik Andreas Rye Berg, Pierre-Marc Jodoin, Thomas Grenier, and others. Deep learning for segmentation using an open large-scale dataset in 2D echocardiography. *IEEE Transactions on Medical Imaging*, 38(9):2198–2210, 2019. Publisher: IEEE.

[88] Jason D Lee, Qi Lei, Nikunj Saunshi, and Jiacheng Zhuo. Predicting what you already know helps: Provable self-supervised learning. *Advances in Neural Information Processing Systems*, 34:309–323, 2021.

[89] Johann Li, Guangming Zhu, Cong Hua, Mingtao Feng, Basheer Bennamoun, Ping Li, Xiaoyuan Lu, Juan Song, Peiyi Shen, Xu Xu, Lin Mei, Liang Zhang, Syed Afaq Ali Shah, and Mohammed Bennamoun. A systematic collection of medical image datasets for deep learning. *ACM Computing Surveys*, 56(5):1–51, May 2024.

[90] Hanying Liang, Guochen Ning, Xinran Zhang, and Hongen Liao. Semi-supervised anatomy tracking with contrastive representation learning in ultrasound sequences. In *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2023.

[91] Daniel Lichtenstein, Ivan Goldstein, Eric Mourgeon, Philippe Cluzel, Philippe Grenier, and Jean-Jacques Rouby. Comparative diagnostic performances of auscultation, chest radiography, and lung ultrasonography in acute respiratory distress syndrome. *The Journal of the American Society of Anesthesiologists*, 100(1):9–15, 2004. Publisher: The American Society of Anesthesiologists.

[92] Daniel A Lichtenstein. *Whole body ultrasonography in the critically ill*. Springer Science & Business Media, 2010.

[93] Daniel A Lichtenstein. Lung ultrasound in the critically ill. *Annals of Intensive Care*, 4:1–12, 2014.

[94] Daniel A Lichtenstein. Current misconceptions in lung ultrasound: A short guide for experts. *Chest*, 156(1):21–25, 2019.

[95] Daniel A Lichtenstein and Yves Menu. A bedside ultrasound sign ruling out pneumothorax in the critically ill: Lung sliding. *Chest*, 108(5):1345–1348, 1995. Publisher: Elsevier.

[96] Daniel A Lichtenstein and Gilbert A Meziere. Relevance of lung ultrasound in the diagnosis of acute respiratory failure: The BLUE protocol. *Chest*, 134(1):117–125, 2008.

[97] Daniel A Lichtenstein, Gilbert Mezière, Nathalie Lascols, Philippe Biderman, Jean-Paul Courret, Agnès Gepner, Ivan Goldstein, and Marc Tenoudji-Cohen. Ultrasound diagnosis of occult pneumothorax. *Critical care medicine*, 33(6):1231–1238, 2005. Publisher: LWW.

[98] Z. Lin, R. Huang, D. Ni, J. Wu, and B. Luo. Masked video modeling with correlation-aware contrastive learning for breast cancer diagnosis in ultrasound. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13543 LNCS:105–114, 2022.

[99] C. Liu, M. Qiao, F. Jiang, Y. Guo, Z. Jin, and Y. Wang. TN-USMA Net: Triple normalization-based gastrointestinal stromal tumors classification on multi-center EUS images with ultrasound-specific pretraining and meta attention. *Medical Physics*, 48(11):7199–7214, 2021.

[100] Hong Liu, Jeff Z HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. *arXiv preprint arXiv:2110.05025*, 2021.

[101] Peizhong Liu, Jiansong Zhang, Xiuming Wu, Shunlan Liu, Yanli Wang, Longxiang Feng, Yong Diao, Zhonghua Liu, Guorong Lyu, and Yongjian Chen. Benchmarking supervised and self-supervised learning methods in a large ultrasound muti-task images dataset. *IEEE Journal of Biomedical and Health Informatics*, 2024.

[102] Shengfeng Liu, Yi Wang, Xin Yang, Baiying Lei, Li Liu, Shawn Xiang Li, Dong Ni, and Tianfu Wang. Deep learning in medical ultrasound analysis: A review. *Engineering*, 5(2):261–275, 2019. Publisher: Elsevier.

[103] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.

[104] H. Lu and Q. Dai. A self-supervised COVID-19 CT recognition system with multiple regularizations. *Computers in Biology and Medicine*, 150, 2022.

[105] Q. Lu, Y. Li, and C. Ye. White matter tract segmentation with self-supervised learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12267 LNCS:270–279, 2020.

[106] Ruben T Lucassen, Mohammad H Jafari, Nicole M Duggan, Nick Jowkar, Alireza Mehrtash, Chanel Fischetti, Denie Bernier, Kira Prentice, Erik P Duhaime, Mike Jin, et al. Deep learning for detection and localization of B-lines in lung ultrasound. *IEEE Journal of Biomedical and Health Informatics*, 27(9):4352–4361, 2023.

[107] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on MRI. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019. Publisher: Elsevier.

[108] D.A. Ma, M.R. Hosseinzadeh Taher, J. Pang, N.U. Islam, F. Haghighi, M.B. Gotway, and J. Liang. Benchmarking and boosting transformers for medical image classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13542 LNCS:12–22, 2022.

[109] Hongxia Ma, Yulin Guo, Qiuping Wang, Min Liu, Yongqian Qiang, Xiaojuan Guo, Youmin Guo, and Qihang Chen. Classification decision tree in CT imaging: Application to the differential diagnosis of solitary pulmonary nodules. *Chinese Journal of Radiology*, pages 50–55, 2008.

[110] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[111] D.M. McSweeney, E.G. Henderson, M. van Herk, J. Weaver, P.A. Bromiley, A. Green, and A. McWilliam. Transfer learning for data-efficient abdominal muscle segmentation with convolutional neural networks. *Medical Physics*, 49(5):3107–3120, 2022.

[112] Oleg V Michailovich and Allen Tannenbaum. Despeckling of medical ultrasound images. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency control*, 53(1):64–78, 2006.

[113] A.K. Mishra, P. Roy, S. Bandyopadhyay, and S.K. Das. CR-SSL: A closely related self-supervised learning based approach for improving breast ultrasound tumor segmentation. *International Journal of Imaging Systems and Technology*, 32(4):1209–1220, 2022.

[114] Silvia Mongodi, Raffaella Arioli, Attilio Quaini, Giuseppina Grugnetti, Anna Maria Grugnetti, and Francesco Mojoli. Lung ultrasound training: How short is too short? Observational study on the effects of a focused theoretical training for novice learners. *BMC Medical Education*, 24(1):166, 2024.

[115] Théo Moutakanni, Maxime Oquab, Marc Szafraniec, Maria Vakalopoulou, and Piotr Bojanowski. You don't need domain-specific data augmentations when scaling self-supervised learning. In *NeurIPS 2024-The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.

[116] Khanjan Nagarsheth and Stanley Kurek. Ultrasound detection of pneumothorax compared with chest X-ray and computed tomography scan. *The American Surgeon*, 77(4):480–483, 2011. Publisher: SAGE Publications Sage CA: Los Angeles, CA.

[117] Peiman Nazerian, Giovanni Volpicelli, Simone Vanni, Chiara Gigli, Laura Betti, Maurizio Bartolucci, Maurizio Zanobetti, Francesca Romana Ermini, Cristina Iannello, and Stefano Grifoni. Accuracy of lung ultrasound for the diagnosis of consolidations when compared to chest computed tomography. *The American Journal of Emergency Medicine*, 33(5):620–625, 2015. Publisher: Elsevier.

[118] Nhut-Quang Nguyen and Thanh-Sach Le. A semi-supervised learning method to remedy the lack of labeled data. In *2021 15th International Conference on Advanced Computing and Applications (ACOMP)*, pages 78–84. IEEE, 2021.

[119] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI*, pages 69–84. Springer, 2016.

[120] C. Ouyang, C. Biffi, C. Chen, T. Kart, H. Qiu, and D. Rueckert. Self-supervision with superpixels: Training few-shot medical image segmentation without annotation.

Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12374 LNCS:762–780, 2020.

[121] David Ouyang, Bryan He, Amirata Ghorbani, Neal Yuan, Joseph Ebinger, Curtis P Langlotz, Paul A Heidenreich, Robert A Harrington, David H Liang, Euan A Ashley, and others. Video-based AI for beat-to-beat assessment of cardiac function. *Nature*, 580(7802):252–256, 2020. Publisher: Nature Publishing Group UK London.

[122] J. Ouyang, Q. Zhao, E. Adeli, G. Zaharchuk, and K.M. Pohl. Self-supervised learning of neighborhood embedding for longitudinal MRI. *Medical Image Analysis*, 82, 2022.

[123] J. Pang, F. Haghighi, D.A. Ma, N.U. Islam, M.R. Hosseinzadeh Taher, M.B. Gotway, and J. Liang. POPAR: Patch order prediction and appearance recovery for self-supervised medical image analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13542 LNCS:77–87, 2022.

[124] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[125] Lina Pedraza, Carlos Vargas, Fabián Narváez, Oscar Durán, Emma Muñoz, and Eduardo Romero. An open access thyroid ultrasound image database. In *10th International Symposium on Medical Information Processing and Analysis*, volume 9287, pages 188–193. SPIE, 2015.

[126] S. Perek, M. Amit, and E. Hexter. Self supervised contrastive learning on multiple breast modalities boosts classification performance. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12928 LNCS:117–127, 2021.

[127] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987. Publisher: Elsevier.

[128] Ross Prager, Hans Clausdorff Fiedler, MMATH Delaney Smith, Derek Wu, and Robert Arntfield. Interrater agreement of physicians identifying lung sliding artifact on B-mode and M-mode point of care ultrasound (POCUS). *POCUS Journal*, 2025.

[129] H. Qi, S. Collins, and J.A. Noble. Knowledge-guided pretext learning for utero-placental interface detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12261 LNCS:582–593, 2020.

[130] Aimon Rahman and Vishal M Patel. UltraMAE: Multi-modal masked autoencoder for ultrasound pre-training. In *Medical Imaging with Deep Learning*, 2024.

[131] Paulo Sergio Rodrigues. Breast ultrasound image. https://data.mendeley.com/datasets/wmy84gzngw/1, 2017.

[132] M. Saeed, R. Muhtaseb, and M. Yaqub. Contrastive pretraining for echocardiography segmentation with limited data. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13413 LNCS:680–691, 2022.

[133] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.

[134] Ashot E Sargsyan, Douglas R Hamilton, Jeffrey A Jones, Shannon Melton, Peggy A Whitson, Andrew W Kirkpatrick, David Martin, and Scott A Dulchavsky. FAST at MACH 20: Clinical ultrasound aboard the International Space Station. *Journal of Trauma and Acute Care Surgery*, 58(1):35–39, 2005.

[135] Madeline C Schiappa, Yogesh S Rawat, and Mubarak Shah. Self-supervised learning for videos: A survey. *ACM Computing Surveys*, 55(13s):1–37, 2023.

[136] Christopher K Schott, Erin Wetherbee, Rahul Khosla, Robert Nathanson, Jason P Williams, Michael J Mader, Elizabeth K Haro, Dean Kellogg III, Abraham Rodriguez, Kevin C Proud, et al. Current use, training, and barriers to point-of-care ultrasound use in ICUs in the Department of Veterans Affairs. *CHEST Critical Care*, 1(2):100012, 2023.

[137] Robert Seifert, Manuel Weber, Emre Kocakavuk, Christoph Rischpler, and David Kersting. Artificial intelligence and machine learning in nuclear medicine: Future perspectives. In *Seminars in Nuclear Medicine*, volume 51, pages 170–177. Elsevier, 2021. Issue: 2.

[138] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[139] M.S. Seyfioğlu, Z. Liu, P. Kamath, S. Gangolli, S. Wang, T. Grabowski, and L. Shapiro. Brain-Aware Replacements for Supervised Contrastive Learning in Detection of Alzheimer's Disease. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13431 LNCS:461–470, 2022.

[140] Eun Jung Shim, Min A Yoon, Hye Jin Yoo, Choong Guen Chee, Min Hee Lee, Sang Hoon Lee, Hye Won Chung, and Myung Jin Shin. An MRI-based decision tree to distinguish lipomas and lipoma variants from well-differentiated liposarcoma of the extremity and superficial trunk: Classification and regression tree (CART) analysis. *European Journal of Radiology*, 127:109012, 2020. Publisher: Elsevier.

[141] Nadya Shusharina, Mattias P Heinrich, and Ruobing Huang. *Segmentation, Classification, and Registration of Multi-modality Medical Imaging Data: MICCAI 2020 Challenge, ABCs 2020, L2R 2020, TN-SCUI 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4–8, 2020, Proceedings*, volume 12587. Springer Nature, 2021.

[142] Prerna Singh, Ramakrishnan Mukundan, and Rex de Ryke. Synthetic models of ultrasound image formation for speckle noise simulation and analysis. In *2017 International Conference on Signals and Systems (ICSigSys)*, pages 278–284. IEEE, 2017.

[143] Nilam J Soni, Robert Arntfield, and Pierre Kory. *Point of care ultrasound e-book*. Elsevier Health Sciences, 2019.

[144] Satoshi Takahashi, Yusuke Sakaguchi, Nobuji Kouno, Ken Takasawa, Kenichi Ishizu, Yu Akagi, Rina Aoyama, Naoki Teraya, Amina Bolatkan, Norio Shinkai, et al. Comparison of vision transformers and convolutional neural networks in medical image analysis: a systematic review. *Journal of Medical Systems*, 48(1):84, 2024.

[145] Aiham Taleb, Winfried Loetzsch, Noel Danz, Julius Severin, Thomas Gaertner, Benjamin Bergner, and Christoph Lippert. 3D self-supervised methods for medical imaging. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors,

Advances in Neural Information Processing Systems, volume 33, pages 18158–18172. Curran Associates, Inc., 2020.

[146] Grace Feng Ling Tan, Tiehua Du, Justin Shuang Liu, Chung Cheen Chai, Chan Maung Nyein, and Allen Yan Lun Liu. Automated lung ultrasound image assessment using artificial intelligence to identify fluid overload in dialysis patients. *BMC Nephrology*, 23(1):410, 2022. Publisher: Springer.

[147] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[148] Franklin N Tessler, William D Middleton, Edward G Grant, Jenny K Hoang, Lincoln L Berland, Sharlene A Teefey, John J Cronan, Michael D Beland, Terry S Desser, Mary C Frates, and others. ACR thyroid imaging, reporting and data system (TI-RADS): white paper of the ACR TI-RADS committee. *Journal of the American College of Radiology*, 14(5):587–595, 2017. Publisher: Elsevier.

[149] Kenneth Thomsen, Lars Iversen, Therese Louise Titlestad, and Ole Winther. Systematic review of machine learning for diagnosis and prognosis in dermatology. *Journal of Dermatological Treatment*, 31(5):496–510, 2020. Publisher: Taylor & Francis.

[150] Ekin Tiu, Ellie Talius, Pujan Patel, Curtis P Langlotz, Andrew Y Ng, and Pranav Rajpurkar. Expert-level detection of pathologies from unannotated chest X-ray images via self-supervised learning. *Nature Biomedical Engineering*, pages 1–8, 2022. Publisher: Nature Publishing Group UK London.

[151] U.S. Food and Drug Administration. Artificial intelligence-enabled device software functions: Lifecycle management and marketing submission recommendations. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/artificial-intelligence-enabled-device-software-functions-lifecycle-management-and-marketing, January 2025. Draft Guidance for Industry and Food and Drug Administration Staff.

[152] Bennett VanBerlo, Delaney Smith, Jared Tschirhart, Blake VanBerlo, Derek Wu, Alex Ford, Joseph McCauley, Benjamin Wu, Rushil Chaudhary, Chintan Dave, and others. Enhancing annotation efficiency with machine learning: Automated partitioning of a lung ultrasound dataset by view. *Diagnostics*, 12(10):2351, 2022. Publisher: MDPI.

[153] Blake VanBerlo, Jesse Hoey, and Alexander Wong. A survey of the impact of self-supervised pretraining for diagnostic tasks in medical X-ray, CT, MRI, and ultrasound. *BMC Medical Imaging*, 24(1):79, 2024. Publisher: Springer.

[154] Blake VanBerlo, Jesse Hoey, Alexander Wong, and Robert Arntfield. The efficacy of semantics-preserving transformations in self-supervised learning for medical ultrasound. *Bioengineering*, 12(8), 2025.

[155] Blake VanBerlo, Brian Li, Jesse Hoey, and Alexander Wong. Self-supervised pretraining improves performance and inference efficiency in multiple lung ultrasound interpretation tasks. *IEEE Access*, 11:135696–135707, 2023. Publisher: IEEE.

[156] Blake VanBerlo, Brian Li, Alexander Wong, Jesse Hoey, and Robert Arntfield. Exploring the utility of self-supervised pretraining strategies for the detection of absent lung sliding in M-mode lung ultrasound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3076–3085, 2023.

[157] Blake VanBerlo, Alexander Wong, Jesse Hoey, and Robert Arntfield. Intra-video positive pairs in self-supervised learning for ultrasound. *Frontiers in Imaging*, 3, 2024.

[158] Blake VanBerlo, Derek Wu, Brian Li, Marwan A Rahman, Gregory Hogg, Bennett VanBerlo, Jared Tschirhart, Alex Ford, Jordan Ho, Joseph McCauley, and others. Accurate assessment of the lung sliding artefact on lung ultrasonography using a deep learning approach. *Computers in Biology and Medicine*, 148:105953, 2022. Publisher: Elsevier.

[159] Gaël Varoquaux and Veronika Cheplygina. Machine learning for medical imaging: methodological failures and recommendations for the future. *NPJ Digital Medicine*, 5(1):48, 2022. Publisher: Nature Publishing Group UK London.

[160] Dominik Vilimek, Jan Kubicek, Milos Golian, Rene Jaros, Radana Kahankova, Pavla Hanzlikova, Daniel Barvik, Alice Krestanova, Marek Penhaker, Martin Cerny, and others. Comparative analysis of wavelet transform filtering systems for noise reduction in ultrasound images. *Plos one*, 17(7):e0270745, 2022. Publisher: Public Library of Science San Francisco, CA USA.

[161] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings*

of the 25th International Conference on Machine Learning, ICML '08, pages 1096–1103, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Helsinki, Finland.

[162] Jian Wang, Xin Yang, Xiaohong Jia, Wufeng Xue, Rusi Chen, Yanlin Chen, Xiliang Zhu, Lian Liu, Yan Cao, Jianqiao Zhou, et al. Thyroid ultrasound diagnosis improvement via multi-view self-supervised learning and two-stage pre-training. *Computers in Biology and Medicine*, 171:108087, 2024.

[163] Y. Wang, D. Song, W. Wang, S. Rao, X. Wang, and M. Wang. Self-supervised learning and semi-supervised learning for multi-sequence medical image classification. *Neurocomputing*, 513:383–394, 2022.

[164] Yu Wang, Xinke Ge, He Ma, Shouliang Qi, Guanjing Zhang, and Yudong Yao. Deep learning in medical ultrasound image analysis: A review. *IEEE Access*, 9:54310–54324, 2021. Publisher: IEEE.

[165] Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945. Publisher: International Biometric Society, Wiley.

[166] Martin J Willemink, Wojciech A Koszek, Cailin Hardell, Jie Wu, Dominik Fleischmann, Hugh Harvey, Les R Folio, Ronald M Summers, Daniel L Rubin, and Matthew P Lungren. Preparing medical imaging data for machine learning. *Radiology*, 295(1):4–15, 2020. Publisher: Radiological Society of North America.

[167] Z. Xiang, Q. Zhuo, C. Zhao, X. Deng, T. Zhu, T. Wang, W. Jiang, and B. Lei. Self-supervised multi-modal fusion network for multi-modal thyroid ultrasound image diagnosis. *Computers in Biology and Medicine*, 150, 2022.

[168] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu. SimMIM: a simple framework for masked image modeling. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9643–9653, Los Alamitos, CA, USA, June 2022. IEEE Computer Society.

[169] Nektaria Xirouchaki, Eleftherios Magkanas, Katerina Vaporidi, Eumorfia Kondili, Maria Plataki, Alexandros Patrianakos, Evaggelia Akoumianaki, and Dimitrios Georgopoulos. Lung ultrasound in critically ill patients: Comparison with bedside chest radiography. *Intensive Care Medicine*, 37:1488–1493, 2011. Publisher: Springer.

[170] Kele Xu, Kang You, Boqing Zhu, Ming Feng, Dawei Feng, and Cheng Yang. Masked modeling-based ultrasound image classification via self-supervised learning. *IEEE Open Journal of Engineering in Medicine and Biology*, 2024.

[171] Zhoubing Xu, Yuankai Huo, JinHyeong Park, Bennett Landman, Andy Milkowski, Sasa Grbic, and Shaohua Zhou. Less is more: Simultaneous view classification and landmark detection for abdominal ultrasound images. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, pages 711–719. Springer, 2018.

[172] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. MedMNIST v2-A large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Scientific Data*, 10(1):41, 2023. Publisher: Nature Publishing Group UK London.

[173] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *IEEE transactions on knowledge and data engineering*, 35(9):8934–8954, 2022.

[174] Yuzhe Yang and Zhi Xu. Rethinking the value of labels for improving class-imbalanced learning. *Advances in Neural Information Processing Systems*, 33:19290–19301, 2020.

[175] Moi Hoon Yap, Gerard Pons, Joan Marti, Sergi Ganau, Melcior Sentis, Reyer Zwiggelaar, Adrian K Davison, and Robert Marti. Automated breast ultrasound lesions detection using convolutional neural networks. *IEEE journal of biomedical and health informatics*, 22(4):1218–1226, 2017. Publisher: IEEE.

[176] Seung Hoon Yoo, Hui Geng, Tin Lok Chiu, Siu Ki Yu, Dae Chul Cho, Jin Heo, Min Sung Choi, Il Hyun Choi, Cong Cung Van, Nguen Viet Nhung, and others. Deep learning-based decision-tree classifier for COVID-19 diagnosis from chest X-ray imaging. *Frontiers in medicine*, 7:427, 2020. Publisher: Frontiers Media SA.

[177] Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training BERT in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

[178] Kathryn Young, Nicole Moon, and Tandi Wilkinson. Building point-of-care ultrasound capacity in rural emergency departments: an educational innovation. *Canadian Journal of Rural Medicine*, 26(4):169–175, 2021.

[179] Maurizio Zanobetti, Claudio Poggioni, and Riccardo Pini. Can chest ultrasonography replace standard chest radiography for evaluation of acute dyspnea in the ED? *Chest*, 139(5):1140–1147, 2011.

[180] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow Twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320, 2021.

[181] E Zhixuan Zeng, Ashkan Ebadi, Adrian Florea, and Alexander Wong. COVID-Net L2C-ULTRA: An explainable linear-convex ultrasound augmentation learning framework to improve COVID-19 assessment and monitoring. *Sensors*, 24(5):1664, 2024. Publisher: MDPI.

[182] Chunhui Zhang, Yixiong Chen, Li Liu, Qiong Liu, and Xi Zhou. HiCo: Hierarchical contrastive learning for ultrasound video model pretraining. In *Proceedings of the Asian Conference on Computer Vision*, pages 229–246, 2022.

[183] Chuyan Zhang, Hao Zheng, and Yun Gu. Dive into the details of self-supervised learning for medical image analysis. *Medical Image Analysis*, 89:102879, 2023. Publisher: Elsevier.

[184] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[185] Kangning Zhang, Jianbo Jiao, and J Alison Noble. Fetal ultrasound video representation learning using contrastive Rubik's Cube recovery. In *International Workshop on Advances in Simplifying Medical Ultrasound*, pages 187–197. Springer, 2024.

[186] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 649–666. Springer, 2016.

[187] Weiwei Zhang, Guang Yang, Nan Zhang, Lei Xu, Xiaoqing Wang, Yanping Zhang, Heye Zhang, Javier Del Ser, and Victor Hugo C de Albuquerque. Multi-task learning with multi-view weighted fusion attention for artery-specific calcification analysis. *Information Fusion*, 71:64–76, 2021. Publisher: Elsevier.

[188] X. Zhang, S. Feng, Y. Zhou, Y. Zhang, and Y. Wang. SAR: Scale-Aware Restoration Learning for 3D Tumor Segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12902 LNCS:124–133, 2021.

[189] Qingyu Zhao, Zixuan Liu, Ehsan Adeli, and Kilian M Pohl. Longitudinal self-supervised learning. *Medical image analysis*, 71:102051, 2021. Publisher: Elsevier.

[190] Z. Zhao and G. Yang. Unsupervised contrastive learning of radiomics and deep features for label-efficient tumor classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12902 LNCS:252–261, 2021.

[191] H. Zheng, J. Han, H. Wang, L. Yang, Z. Zhao, C. Wang, and D.Z. Chen. Hierarchical self-supervised learning for medical image segmentation based on multi-domain data aggregation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12901 LNCS:622–632, 2021.

[192] Ruifeng Zheng, Ying Zhong, Senxiang Yan, Hongcheng Sun, Haibin Shen, and Kejie Huang. MsVRL: Self-supervised multiscale visual representation learning via cross-level consistency for medical image segmentation. *IEEE Transactions on Medical Imaging*, 42(1):91–102, 2023.

[193] Yi Zheng, Rushin H Gindra, Emily J Green, Eric J Burks, Margrit Betke, Jennifer E Beane, and Vijaya B Kolachalama. A graph-transformer for whole slide image classification. *IEEE Transactions on Medical Imaging*, 41(11):3003–3015, 2022. Publisher: IEEE.

[194] Z. Zhou, V. Sodha, J. Pang, M.B. Gotway, and J. Liang. Models Genesis. *Medical Image Analysis*, 67, 2021.

[195] Z. Zhou, C. Zhao, H. Qiao, M. Wang, Y. Guo, Q. Wang, R. Zhang, H. Wu, F. Dong, Z. Qi, J. Li, X. Tian, X. Zeng, Y. Jiang, F. Xu, Q. Dai, and M. Yang. RATING: Medical knowledge-guided rheumatoid arthritis assessment from multimodal ultrasound images via deep learning. *Patterns*, 3(10), 2022.

[196] J. Zhu, Y. Li, Y. Hu, K. Ma, S.K. Zhou, and Y. Zheng. Rubik's Cube+: A self-supervised feature learning framework for 3D medical image analysis. *Medical Image Analysis*, 64, 2020.

[197] R. Zoetmulder, E. Gavves, M. Caan, and H. Marquering. Domain- and task-specific transfer learning for medical segmentation tasks. *Computer Methods and Programs in Biomedicine*, 214, 2022.

APPENDICES

Appendix A

Private Dataset Access

The private datasets in Chapters 3, 4, 5, and 6 were assembled from a database of lung ultrasound examinations that were retrospectively collected during routine clinical care. As part of a larger study that received approval from the Research Ethics Board (REB) at Western University (approval # 116838), these examinations were de-identified and compiled into a research dataset. Deep Breathe removed burnt-in graphical elements from the ultrasound videos in the dataset and annotated a subset of them. Several videos remain unannotated.

A data transfer agreement permitted the dataset to be shared with the University of Waterloo, where the study also received REB approval (approval # 43986). The research comprising this thesis is included within that study.

A sponsored research agreement (SRA # 102082) between the University of Waterloo and Deep Breathe provided the author with access to the dataset for the research detailed in this thesis. All data access and usage complied with the applicable REB approvals, the data transfer agreement, and the terms of the SRA.

Appendix B

Additional Classification Metrics for Chapter 3

Tables B.1 and B.2 provide additional classification metrics from the evaluation on the local and external test sets, respectively (detailed in Sections 3.4.1, 3.4.2, and 3.4.3). To supplement the AUC reported in Table 3.3, we provide precision, recall (i.e., sensitivity), and specificity. All available training labels were employed for these experiments.

Table B.1 provides a more detailed picture of the performance on the local test set. Notably, metrics are similar for `View`, with most pretrained models exhibiting slightly greater performance in the fine-tuning setting for each metric. For the `AB` task, pretrained models exhibited decidedly greater precision and specificity, but lower recall. The fully supervised models predicted very few false positives for `PE` on the local test set, as evidenced by their high precision and specificity. However, the recall of linear and fine-tuned `PE` classifiers was abhorrently low, suggesting the presence of a plenitude of false negative predictions. The fully supervised MLP trained for `PE` performed particularly well compared to pretrained models.

External test set metrics are reported in Table B.2. In contrast to the local test set, the performance of pretrained models was generally better than that of fully supervised models across all three tasks. Most notably, linear classifiers always achieved greater precision, recall, and specificity on external test data. The findings indicated that feature representations from pretrained models may be more generalizable than fully supervised models.

Task	Pretraining	Precision			Recall			Specificity		
		LC	FT	NC	LC	FT	NC	LC	FT	NC
`View`	SimCLR	0.922	0.930	0.932	0.855	0.921	0.946	0.968	0.969	0.972
	Barlow Twins	0.908	0.928	0.919	0.838	0.909	0.887	0.963	0.969	0.966
	VICReg	0.896	0.929	0.877	0.231	0.903	0.877	0.958	0.970	0.964
	None	0.913	0.912	0.916	0.799	0.906	0.879	0.967	0.961	0.965
`AB`	SimCLR	0.893	0.934	0.902	0.795	0.839	0.815	0.961	0.976	0.964
	Barlow Twins	0.812	0.872	0.857	0.799	0.818	0.793	0.925	0.951	0.946
	VICReg	0.827	0.859	0.825	0.792	0.819	0.774	0.932	0.945	0.933
	None	0.852	0.778	0.885	0.677	0.861	0.760	0.952	0.899	0.960
`PE`	SimCLR	0.824	0.908	0.848	0.593	0.680	0.662	0.941	0.968	0.945
	Barlow Twins	0.801	0.899	0.841	0.556	0.699	0.650	0.935	0.963	0.943
	VICReg	0.838	0.928	0.817	0.526	0.717	0.602	0.953	0.974	0.937
	None	0.925	0.996	0.940	0.468	0.371	0.667	0.982	0.999	0.980

Table B.1: Classification metrics calculated based on predictions for the local test set for the LC, FT, and NC experiments.

Task	Pretraining	Precision			Recall			Specificity		
		LC	FT	NC	LC	FT	NC	LC	FT	NC
`View`	SimCLR	0.875	0.877	0.889	0.737	0.791	0.762	0.963	0.961	0.966
	Barlow Twins	0.892	0.931	0.930	0.725	0.769	0.724	0.969	0.980	0.981
	VICReg	0.946	0.918	0.926	0.672	0.771	0.720	0.986	0.976	0.980
	None	0.783	0.846	0.811	0.664	0.769	0.737	0.935	0.951	0.939
`AB`	SimCLR	0.906	0.866	0.876	0.676	0.736	0.729	0.929	0.885	0.896
	Barlow Twins	0.931	0.904	0.901	0.704	0.742	0.726	0.948	0.921	0.920
	VICReg	0.935	0.921	0.928	0.684	0.726	0.698	0.952	0.937	0.945
	None	0.810	0.730	0.847	0.612	0.830	0.665	0.855	0.691	0.880
`PE`	SimCLR	0.979	0.995	0.967	0.663	0.643	0.627	0.993	0.998	0.989
	Barlow Twins	0.936	0.996	0.984	0.588	0.605	0.615	0.979	0.999	0.995
	VICReg	0.876	0.998	0.982	0.618	0.602	0.649	0.954	0.999	0.994
	None	0.739	1.000	0.999	0.482	0.235	0.597	0.910	1.000	1.000

Table B.2: Classification metrics calculated based on predictions for the external test set for the LC, FT, and NC experiments.

Appendix C

Self-supervised Learning Objectives with Sample Weights

Sample weights were incorporated into the objective functions of each self-supervised learning method employed in Chapter 5: SimCLR [29], VICReg [12], and Barlow Twins [180]. As outlined in Section 5.3.1, sample weights $w_i \in \left[\frac{1}{\delta+1}, 1\right]$ were inversely proportional to the temporal or spatial distance between intra-video positive pairs. In this appendix, we propose weighted versions of the SimCLR, VICReg, and Barlow Twins objective functions.

C.1 Weighted SimCLR Objective

The SimCLR objective can be easily modified by multiplying the per-example NT-Xent loss for the i^{th} positive pair L_i by sample weight w_i.

$$\mathcal{L}_{\text{SimCLR}} = \frac{1}{N} \sum_{i=1}^{N} w_i L_i \tag{C.1}$$

C.2 Weighted VICReg Objective

For VICReg [12], the invariance term is weighted with w_i for each positive pair in a batch. The invariance term is then calculated as follows:

$$s(Z_1, Z_2) = \frac{1}{N} \sum_{i=1}^{N} w_i \|Z_{1_i} - Z_{2_j}\|_2^2 \tag{C.2}$$

where Z_1 and Z_2 are batches of predicted embeddings for corresponding positive pairs; that is, Z_{1i} and Z_{2i} correspond to one positive pair. The entire VICReg objective can then be calculated as

$$\mathcal{L}_{\text{VICReg}}(Z_1, Z_2) = \underbrace{\lambda s(Z_1, Z_2)}_{\text{Invariance term}} + \underbrace{\mu(v(Z_1) + v(Z_2))}_{\text{Variance term}} + \underbrace{\nu(c(Z_1) + c(Z_2))}_{\text{Covariance term}} \tag{C.3}$$

where λ, μ, and ν are weights for each term. Since frames are sampled uniformly at random, $\mathbb{E}[w] \simeq 0.5$. Accordingly, we doubled λ when pretraining VICReg with sample weights to maintain the overall scale of the invariance term.

C.3 Weighted Barlow Twins Objective

For the Barlow Twins objective [180], weighting is applied to each positive pair in the invariance term by computing the weighted normalized cross-correlation matrix $\mathcal{C}_W \in \mathbb{R}^{D \times D}$ between the weighted-mean-centred normalized batches of embeddings, Z_1 and Z_2. For a batch of embeddings Z, the calculation for the weighted mean $\bar{Z}_W$ and standard deviations $\sigma_W(Z)$ across the batch dimension is performed as follows:

$$\bar{Z}_W = \frac{\sum_{i=1}^{N} w_i Z_i}{\sum_{i=1}^{N} w_i} \tag{C.4}$$

$$\sigma_W(Z) = \sqrt{\frac{\sum_{i=1}^{N} w_i (Z_i - \bar{Z})^2}{\sum_{i=1}^{N} w_i}} \tag{C.5}$$

where w_i is the sample weight for the i^{th} positive pair in the batch. In the original Barlow Twins method, the (unweighted) normalized cross-correlation matrix is computed using the batch-wise embedding means $\bar{Z}_1$ and $\bar{Z}_2$, along with the batch-wise standard deviations $\sigma(Z_1)$ and $\sigma(Z_2)$. Equations C.6 and C.7 show the calculation of the unweighted and weighted normalized cross-correlation matrices, respectively.

$$\mathcal{C} = \frac{1}{N}\left(\frac{Z_1 - \bar{Z}_1}{\sigma(Z_1)}\right)^T \left(\frac{Z_2 - \bar{Z}_2}{\sigma(Z_2)}\right) \tag{C.6}$$

$$\mathcal{C}_W = \frac{1}{N}\left(\frac{Z_1 - \bar{Z}_{1W}}{\sigma_W(Z_1)}\right)^T \left(\frac{Z_2 - \bar{Z}_{2W}}{\sigma_W(Z_2)}\right) \tag{C.7}$$

The invariance term of the Barlow Twins objective is computed using $\mathcal{C}_W$. The redundancy reduction term employs $\mathcal{C}$ since its purpose is to decorrelate the embedding dimensions. In the original Barlow Twins method, $\mathcal{C}$ is used for both terms. Equation C.8 gives the weighted Barlow Twins objective.

$$\mathcal{L}_{\text{BT}} = \underbrace{\sum_{d=1}^{D}(1 - \mathcal{C}_{W_{d,d}})^2}_{\text{Invariance term}} + \underbrace{\lambda \sum_{d=1}^{D}\sum_{\substack{e=1 \\ e\neq d}}^{D} {\mathcal{C}_{d,e}}^2}_{\text{Redundancy reduction term}} \tag{C.8}$$

Appendix D

Intra-Video Positive Pair Results

In Section 5.4.2, results derived from varying Intra-Video Positive Pairs hyperparameters are displayed in charts. The same results are reported in tables here, for clarity. Average test accuracy for all values of δ_t with and without sample weights are given in Table D.1. Table D.2 details the performance of linear classifiers trained on representations outputted by a pretrained feature extractor. Lastly, Table D.3 provides the results of fine-tuning pretrained feature extractors. The metrics provided in these tables are the same as those conveyed in Figures 5.3 and 5.4 as bar charts.

Pretraining Method	δ_t	SW	Mean (std) Test Accuracy
SimCLR	0	✗	0.919 (0.041)
	0.5	✗	0.926 (0.043)
	0.5	✓	0.922 (0.028)
	1	✗	0.911 (0.049)
	1	✓	0.916 (0.041)
	1.5	✗	0.911 (0.047)
	1.5	✓	0.924 (0.042)
Barlow Twins	0	✗	0.905 (0.061)
	0.5	✗	0.912 (0.048)
	0.5	✓	0.904 (0.052)
	1	✗	0.914 (0.057)
	1	✓	0.911 (0.052)
	1.5	✗	0.921 (0.054)
	1.5	✓	0.910 (0.041)
VICReg	0	✗	0.917(0.046)
	0.5	✗	0.924 (0.044)
	0.5	✓	0.912 (0.050)
	1	✗	0.925 (0.052)
	1	✓	0.930 (0.046)
	1.5	✗	0.913 (0.045)
	1.5	✓	0.918 (0.035)
None (ImageNet-pretrained)			0.908 (0.043)
None (random initialization)			0.881 (0.050)

Table D.1: Mean 5-fold cross validation test accuracy for ResNet18 classifiers pretrained on the Butterfly dataset and fine-tuned on the POCUSBorn dataset.

			AB			LS
Pretraining Method	δ_t	SW	Test AUC	δ_x	SW	Test AUC
SimCLR	0	✗	0.952	0	✗	0.863
	0.5	✗	0.962	5	✗	0.867
	0.5	✓	0.961	5	✓	0.874
	1	✗	0.964	10	✗	0.865
	1	✓	0.963	10	✓	0.872
	1.5	✗	0.958	15	✗	0.864
	1.5	✓	0.959	15	✓	0.860
Barlow Twins	0	✗	0.903	0	✗	0.818
	0.5	✗	0.922	5	✗	0.832
	0.5	✓	0.923	5	✓	0.853
	1	✗	0.914	10	✗	0.834
	1	✓	0.915	10	✓	0.849
	1.5	✗	0.906	15	✗	0.838
	1.5	✓	0.905	15	✓	0.856
VICReg	0	✗	0.938	0	✗	0.830
	0.5	✗	0.912	5	✗	0.853
	0.5	✓	0.921	5	✓	0.829
	1	✗	0.907	10	✗	0.843
	1	✓	0.904	10	✓	0.838
	1.5	✗	0.916	15	✗	0.834
	1.5	✓	0.9135	15	✓	0.837
None (ImageNet-pretrained)			0.906			0.857
None (random initialization)			0.604			0.516

Table D.2: ParenchymalLUS test AUC for linear classifiers trained on the AB and LS tasks when trained on all examples in the training set.

			AB			LS
Pretraining Method	δ_t	SW	Test AUC	δ_x	SW	Test AUC
SimCLR	0	✗	0.977	0	✗	0.893
	0.5	✗	0.977	5	✗	0.893
	0.5	✓	0.978	5	✓	0.893
	1	✗	0.977	10	✗	0.897
	1	✓	0.979	10	✓	0.898
	1.5	✗	0.980	15	✗	0.903
	1.5	✓	0.980	15	✓	0.887
Barlow Twins	0	✗	0.969	0	✗	0.867
	0.5	✗	0.968	5	✗	0.870
	0.5	✓	0.965	5	✓	0.882
	1	✗	0.966	10	✗	0.863
	1	✓	0.964	10	✓	0.879
	1.5	✗	0.965	15	✗	0.887
	1.5	✓	0.966	15	✓	0.886
VICReg	0	✗	0.971	0	✗	0.862
	0.5	✗	0.965	5	✗	0.861
	0.5	✓	0.963	5	✓	0.862
	1	✗	0.962	10	✗	0.856
	1	✓	0.966	10	✓	0.859
	1.5	✗	0.969	15	✗	0.856
	1.5	✓	0.967	15	✓	0.861
None (ImageNet-pretrained)			0.973			0.898
None (random initialization)			0.954			0.790

Table D.3: ParenchymalLUS test AUC for classifiers fine-tuned on the AB and LS tasks when trained on all examples in the training set.

Appendix E

Batch Size Robustness for Intra-Video Positive Pairs

Batch size has previously been reported as a salient hyperparameter for SimCLR, as its authors found that larger batch sizes tended to yield better representations [29]. Indeed, one of the motivations behind the development of non-contrastive methods was to mitigate the apparent need for large batch sizes. The original VICReg study found that VICReg and Barlow Twins exhibited greatly lesser performance drops with smaller batch sizes [12]. A recent study found that SimCLR can also be more robust to smaller batch sizes, depending on alternate hyperparameter assignments and the nature of the downstream task [18].

To explore whether batch size impacts the quality of representations produced with IVPP pretraining, we compared the performance of classifiers for the `AB` task pretrained on ParenchymalLUS using batch sizes of 256, 384, and 512. We pretrained feature extractors using IVPP with sample weights and $\delta_t = 1$ for each of the SimCLR, Barlow Twins, and VICReg objectives. Both linear and fine-tuned classifiers were trained, according to the protocols described in Section 5.4.1. Table E.1 displays the results. Test AUC remained similar for all classifiers trained with SimCLR and Barlow Twins across all three pretraining batch sizes. VICReg's performance was nearly constant for smaller pretraining batch sizes; however, the pretrained model with a batch size of 512 experienced collapse, which is reflected in the downstream task performance. Optimization of other VICReg hyperparameters may mitigate the risk of collapse [12].

Pretraining Method	Batch Size	Test AUC	
		Linear classifier	Fine-tuned classifier
SimCLR	256	0.965	0.977
	384	0.963	0.979
	512	0.963	0.977
Barlow Twins	256	0.905	0.964
	384	0.915	0.964
	512	0.914	0.968
VICReg	256	0.907	0.966
	384	0.904	0.966
	512	0.500	0.500

Table E.1: Performance of linear classifiers and fine-tuned classifier trained on the `AB` task with various values for the pretraining batch size.

Appendix F

Data Augmentation Details

This section provides descriptions and algorithmic details for the StandardAug and AugUS-O data augmentation pipelines discussed in Section 6.3.3. Transformations are referred to both by name and by the identifiers given in Tables 6.4 and 6.5.

F.1 StandardAug Transformations

We investigated a standard data augmentation pipeline that has used extensively in the SSL literature [12, 29, 59, 180]. To standardize experiments, we adopted the symmetric version from the VICReg paper [12], which uses the same transformations and probabilities for each of the two branches in the joint embedding architecture. The transformations and their parameter settings are widely adopted. Below, we detail their operation and parameter assignments.

F.1.1 Crop and Resize (B00)

A rectangular crop of the input image is designated at a random location. The area of the cropped region is sampled from the uniform distribution $\mathcal{U}(0.08, 1)$. The cropped region's aspect ratio is sampled from the uniform distribution $\mathcal{U}(0.75, 1.33)$. Its width and height are calculated accordingly. The cropped region is then resized to the original image dimension.

F.1.2 Horizontal Reflection (B01)

The image is reflected about the central vertical axis.

F.1.3 Color Jitter (B02)

The brightness, contrast, saturation, and hue of the image are modified. The brightness change factor, contrast change factor, saturation change factor, and hue change factor are sampled from $\mathcal{U}(0.6, 1.4)$, $\mathcal{U}(0.6, 1.4)$, $\mathcal{U}(0.8, 1.2)$, and $\mathcal{U}(-0.1, 0.1)$, respectively.

F.1.4 Conversion to Grayscale (B03)

Images are converted to grayscale. The output images have three channels, such that each channel has the same pixel intensity.

F.1.5 Gaussian Blur (B04)

The image is denoised using a Gaussian blur with kernel size 13 and standard deviation sampled uniformly at random from $\mathcal{U}(0.1, 2)$. Note that in the original pipeline, the kernel size was set to 23 and 224×224 images were used. We used 128×128 images; as such, we selected a kernel size that covers a similar fraction of the image.

F.1.6 Solarization (B05)

All pixels with intensity of 128 or greater are inverted. Note that the inputs are unsigned 8-bit images.

F.2 Ultrasound-Specific Transformations

In this section, we provide details on the set of transformations that comprise AugUS-v1. Several of the transformations operate on the pixels contained within the ultrasound field of view (FOV). As such, the geometrical form of the FOV was required to perform some transformations. We adopted the same naming convention for the vertices of the

ultrasound FOV as Kim *et al.* [85]. Let p_1, p_2, p_3, and p_4 represent the respective locations of the top left, top right, bottom left, and bottom right vertices, and let $\langle x_i, y_i \rangle$ be the x- and y-coordinates of p_i in image space. For convex FOV shapes, we denote the intersection of lines $\overleftrightarrow{p_1 p_3}$ and $\overleftrightarrow{p_2 p_4}$ as p_0. Figure F.1 depicts the arrangement of these points for each of the three main US FOV shapes: linear, curvilinear, and phased array. A software tool was used to estimate the FOV vertices and probe type for all videos in each dataset (UltraMask, Deep Breathe Inc., Canada).

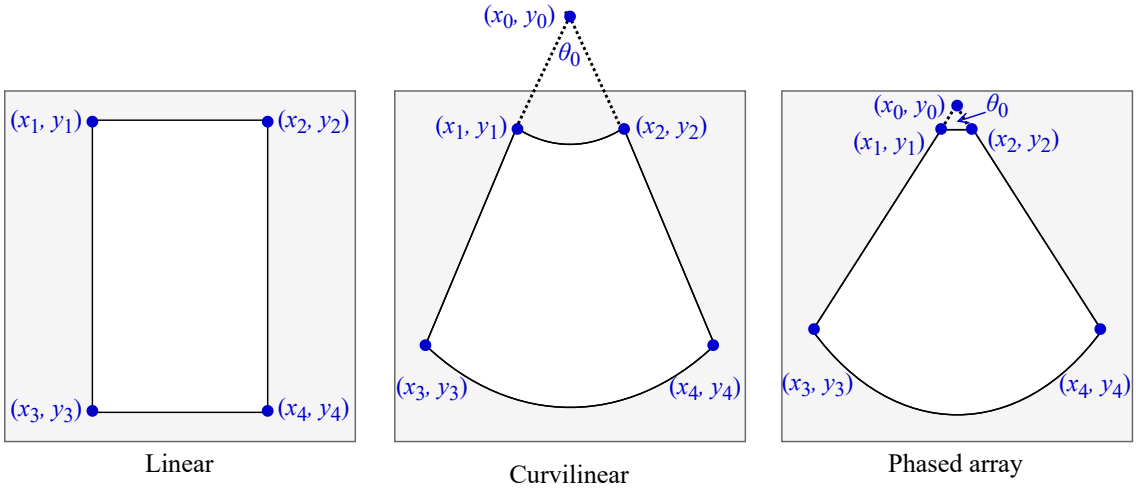


Figure F.1: Locations of the named FOV vertices for each of the three main field of view shapes in US imaging

F.2.1 Probe Type Change (U00)

To produce a transformed ultrasound image with a different field of view (FOV) shape, a mapping that gives the location of pixels in the original image for each coordinate in the new image is calculated. Concretely, the function $\mathbf{f} : \mathbb{R}^2 \to [-1, 1]^2$ returns the coordinates of the point in the original image that corresponds to a point in the transformed image. Note that $(-1, -1)$ corresponds to the top left of the original image. Pixel intensities in the transformed image are interpolated according to their corresponding location in the original image.

Algorithm 1 details the calculation of $\mathbf{f}_{\ell \to c}$ for converting linear FOV to convex FOVs with a random radius factor $\rho \sim \mathcal{U}(1, 2)$, along with new FOV vertices. Similarly, curvilin-

ear and phased array FOV shapes are converted to linear FOV shapes. Algorithm 2 details the calculation of the mapping $\mathbf{f}_{c\to\ell}$ that transforms convex FOV shapes to linear shapes, along with calculations for the updated named coordinates. To ensure that no aspects of the old FOV remain on the image, bitmask $M' \in \{0,1\}^{h\times w}$ is produced using the new named coordinates.

Since the private dataset was resized to square images that exactly encapsulated the FOV, images are resized to match their original aspect ratios to ensure that the sectors were circular. They are then resized to their original dimensions following the transformation.

Algorithm 1 Compute a point mapping for linear to curvilinear FOV shape, along with new FOV vertices

Require: FOV vertices p_1, p_2, p_3, p_4; radius factor ρ; coordinate maps $\mathbf{x} = \mathbf{1}_{h\times 1}[0, 1, \ldots, w-1]$ and $\mathbf{y} = [0, 1, \ldots, h-1]^T\mathbf{1}_{1\times w}$

$r_b \leftarrow \rho(y_3 - y_1)$ ▷ *Bottom sector radius*

$x'_0 \leftarrow \max(x_3, 0) + (x_4 - x_3)/2$

$y'_0 \leftarrow y_3 - r_b$ ▷ *Lateral bounds will intersect at* (x'_0, y'_0)

$y'_1 = y'_2 \leftarrow y_1$

$y'_3 = y'_4 \leftarrow y'_0 + \sqrt{r_b^2 - (x'_0 - x_1)^2}$

$x'_1 \leftarrow x'_0 - (y_1 - y'_0)(x'_0 - x_3)/(y'_3 - y'_0)$

$x'_2 \leftarrow x'_0 + (y_1 - y'_0)(x'_0 - x_3)/(y'_3 - y'_0)$

$r_t \leftarrow \sqrt{(x'_0 - x'_1)^2 + (y_1 - y'_0)^2}$ ▷ *Top sector radius*

$\boldsymbol{\phi} \leftarrow \operatorname{atan2}(\mathbf{x} - x'_0, \mathbf{y} - y'_0)$ ▷ *Angle with the central vertical*

$$\mathbf{f}_{\ell\to c} \leftarrow \begin{pmatrix} \frac{\boldsymbol{\phi} + (x_i - w/2)/w}{|\,\boldsymbol{\phi}[y'_3, 0]\,|} \\ 2\frac{\sqrt{(x'_0 - \mathbf{x})^2 + (y'_0 - \mathbf{y})^2} - r_t}{r_b - r_t} - 1 \end{pmatrix}$$ ▷ *Final coordinate mapping*

return $\mathbf{f}_{\ell\to c}$, p'_1, p'_2, p'_3, p'_4 ▷ *Coordinate mapping, new FOV vertices*

F.2.2 Convexity Change (U01)

To mimic an alternative convex FOV shape with a different θ_0, a mapping is calculated that results in a new FOV shape wherein p_0 is translated vertically. A new value for the width of the top of the FOV is randomly calculated, facilitating the specification of a new p_0. Given the new p_0, a pixel map $\mathbf{f}_{c\to c'}$ is computed according to Algorithm 3. Similar to the

Algorithm 2 Compute a point mapping for convex to linear FOV shape, along with new FOV vertices

Require: FOV vertices p_1, p_2, p_3, p_4; point of intersection p_0; angle of intersection θ_0; width fraction $\omega \in [0, 1]$; coordinate maps $\mathbf{x} = \mathbf{1}_{h\times 1}[0, 1, \ldots, w-1]$ and $\mathbf{y} = [0, 1, \ldots, h-1]^T \mathbf{1}_{1\times w}$

$r_b \leftarrow \sqrt{(x_0 - x_3)^2 + (y_0 - y_3)^2}$ ▷ *Bottom sector radius*

$x'_1 = x'_3 \leftarrow x_0 - \omega w/2$

$x'_2 = x'_4 \leftarrow x_0 + \omega w/2$

$y'_1 = y'_2 \leftarrow y_1$

$y'_3 = y'_4 \leftarrow y_0 + r_b$

$\boldsymbol{\phi} \leftarrow \theta_0((\mathbf{x} - x_3)/(x_4 - x_3) - \frac{1}{2})$ ▷ *Angle with the central vertical*

$\mathbf{y_n} \leftarrow (\mathbf{y} - y_1)/(y_0 + r_b - y_1)$ ▷ *Normalized y-coordinates*

if probe type is curvilinear **then**

  $r_t \leftarrow \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}$ ▷ *Curvilinear; top sector radius*

else

  $r_t \leftarrow (y_1 - y_0)/\cos(\boldsymbol{\phi}/w)$ ▷ *Phased array; distance to top bound*

end if

$$\mathbf{f}_{c\to\ell} \leftarrow \begin{pmatrix} x_0 + \sin(\boldsymbol{\phi}/w)(r_t + \mathbf{y_n}(r_b - r_t)) \\ y_0 + \cos(\boldsymbol{\phi}/w)(r_t + \mathbf{y_n}(r_b - r_t)) \end{pmatrix}$$ ▷ *Final coordinate mapping*

return $\mathbf{f}_{c\to\ell}$, p'_1, p'_2, p'_3, p'_4 ▷ *Coordinate mapping, new FOV vertices*

probe type change transformation, pixel intensities at each coordinate in the transformed image are interpolated according to the corresponding coordinate in the original image returned by $\mathbf{f}_{c \to c'}$.

Algorithm 3 Compute a point mapping from an original to a modified convex FOV shape.

Require: FOV vertices p_1, p_2, p_3, p_4; point of intersection p_0; angle of intersection θ_0; new top width w'; coordinate maps $\mathbf{x} = \mathbf{1}_{h \times 1}[0, 1, \ldots, w-1]$ and $\mathbf{y} = [0, 1, \ldots, h-1]^T \mathbf{1}_{1 \times w}$

$s \leftarrow w'(x_4 - x_3)/(x_2 - x_1)$ ▷ *Scale change for top bound*
$x'_1 \leftarrow x_0 - s(x_0 - x_1)$
$x'_2 \leftarrow x_0 + s(x_2 - x_0)$
$y'_1 = y'_2 \leftarrow y_1$
$y'_3 = y'_4 \leftarrow y_3$
$\theta'_0, p'_0 \leftarrow$ angle, point of intersection of $\overleftrightarrow{p'_1 p'_3}$ and $\overleftrightarrow{p'_2 p'_4}$
$r_b \leftarrow \sqrt{(x_0 - x_3)^2 + (y_0 - y_3)^2}$ ▷ *Current bottom radius*
$r'_b \leftarrow \sqrt{(x'_0 - x'_3)^2 + (y'_0 - y'_3)^2}$ ▷ *New bottom radius*
$r_t \leftarrow \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}$ ▷ *Current top radius*
$r'_t \leftarrow \sqrt{(x'_0 - x'_1)^2 + (y'_0 - y'_1)^2}$ ▷ *New top radius*
$\boldsymbol{\phi}' \leftarrow \text{atan2}(\mathbf{x} - x'_0, \mathbf{y} - y'_0)$
$\mathbf{r}' \leftarrow \sqrt{(x'_0 - \mathbf{x})^2 + (y'_0 - \mathbf{y})^2}$
$\mathbf{r}' \leftarrow (\mathbf{r}' - r'_t)(r_b - r_t)/(r'_b - r'_t) + r_t$
$\mathbf{f}_{c \to c'}(x, y) \leftarrow \begin{pmatrix} x_0 + \mathbf{r}' \sin(\boldsymbol{\phi}' \theta_0 / \theta'_0) \\ y_0 + \mathbf{r}' \cos(\boldsymbol{\phi}' \theta_0 / \theta'_0) \end{pmatrix}$
return $\mathbf{f}_{c \to c'}$, p'_1, p'_2, p'_3, p'_4 ▷ *Coordinate mapping, new FOV vertices*

F.2.3 Wavelet Denoising (U02)

Following the soft thresholding method by Birgé and Massart [15], we apply a wavelet transform, conduct thresholding, then apply the inverse wavelet transform. The mother wavelet is randomly chosen from a set, and the sparsity parameter α is sampled from a uniform distribution. We use $J = 3$ levels of wavelet decomposition and set the decomposition level $J_0 = 2$. We designated Daubechies wavelets $\{db2, db5\}$ as the set of mother

wavelets, which is a subset of those identified by Vilimek *et al.*'s assessment [160] as most suitable for denoising US images.

F.2.4 Contrast-Limited Adaptive Histogram Equalization (U03)

Contrast-limited adaptive histogram equalization is applied to the input image. The transformation enhances low-contrast regions of ultrasound images while avoiding excessive noise amplification. We found that CLAHE enhances artifact The tiles are 8×8 regions of pixels. The clip limit is sampled from the uniform distribution $\mathcal{U}(30, 50)$.

F.2.5 Gamma Correction (U04)

The pixel intensities of the image are nonlinearly modified. Pixel intensity I is transformed as follows:

$$I' \leftarrow 255 \left(\frac{I}{255}\right)^{\gamma} \tag{F.1}$$

where $\gamma \sim \mathcal{U}(0.5, 1.75)$. The gain is fixed at 1.

F.2.6 Brightness and Contrast Change (U05)

The brightness and contrast of the image are modified. The brightness change factor, contrast change factor are sampled from $\mathcal{U}(0.6, 1.4)$, $\mathcal{U}(0.6, 1.4)$, respectively. The image is then multiplied element-wise by its FOV mask, to ensure black regions external to the FOV remain black.

F.2.7 Depth Change Simulation (U06)

The image is zoomed about a point that differs according to FOV type, simulating a change in imaging depth. The transformation preserves the centre for linear FOV shapes and preserves p_0 for convex FOV shapes. The magnitude of the zoom transformation, d, is randomly sampled from a uniform distribution. Increasing the depth corresponds to zooming out ($d > 1$), while decreasing the depth corresponds to zooming in ($d < 1$).

F.2.8 Speckle Noise Simulation (U07)

Following Singh *et al.*'s method [142], this transformation calculates synthetic speckle noise and applies it to the ultrasound FOV. Various parameters of the algorithm are randomly sampled upon each invocation. The lateral and axial resolutions for interpolation are random integers drawn from the ranges $[35, 45]$ and $[75, 85]$, respectively. The number of synthetic phasors is randomly drawn from the integer range $[5, 10]$. Sample points on the image are evenly spaced in Cartesian coordinates for linear FOV shapes. For convex FOVs, the sample points are evenly spaced in polar coordinates.

F.2.9 Gaussian Noise Simulation (U08)

Multiplicative Gaussian noise is applied to the pixel intensities across the image. First, the standard deviation of the Gaussian noise, σ, is randomly drawn from the uniform distribution $\mathcal{U}(0.5, 2.5)$. Multiplicative Gaussian noise with mean 1 and standard deviation σ is then applied independently to each pixel in the image.

F.2.10 Salt and Pepper Noise Simulation (U09)

A random assortment of points in the image are set to 255 (salt) or 0 (pepper). The fractions of pixels set to salt and pepper values are sampled randomly according to $\mathcal{U}(0.001, 0.005)$.

F.2.11 Horizontal Reflection (U10)

The image is reflected about the central vertical axis. This transformation is identical to U01.

F.2.12 Rotation and Shift (U11)

A non-scaling affine transformation is applied to the image. More specifically, the image is translated and rotated. The horizontal component of the translation is sampled from $\mathcal{U}(-0.2, 0.2)$, as a fraction of the image's width. Similarly, the vertical component is sampled from $\mathcal{U}(-0.2, 0.2)$, as a fraction of the image's height. The rotation angle, in degrees, is sampled from $\mathcal{U}(-22.5, 22.5)$.

Appendix G

Pleural Line Object Detection Training

The Single Shot Detector (SSD) method [103] was employed to perform a cursory evaluation of the pretrained models on an object detection task. The `PL` task involved localizing instances of the pleural line in lung US images, which can be described as a bright horizontal line that is typically situated slightly below the vertical level of the ribs. It is only visible between the rib spaces since bone obstructs ultrasound waves. The artifact represents the interface between the parietal and visceral linings of the lung.

As in the classification experiments, we used the MobileNetV3Small architecture as the backbone of the network [71]. There is precedent for using the SSD object detection method, as it has been applied to assess the object detection capabilities of MobileNet architectures [71, 133]. The feature maps outputted from a designated set of layers were passed to the SSD regression and classification heads. We selected a range of layers whose feature maps had varying spatial resolution. Table G.1 provides the identities and dimensions of the feature maps from the backbone that were fed to the SSD model heads. The head contained 43 796 trainable parameters, which was light compared to the backbone (927 008).

The set of default anchor box aspect ratios was manually specified after examining the distribution of bounding box aspect ratios in the training set. The 25^{th} percentile was 2.894, and the 75^{th} percentile was 4.989. The pleural line artifact typically has a much greater width than height. Accordingly, we designated the set of default anchor box aspect ratios (w/h) as $\{1, 2, 3, 4, 5\}$. Six anchor box scales were used. The first five were spaced out evenly over the range $[0.023, 0.170]$, which correspond to the square roots of

the minimum and maximum areas of the bounding box labels present in the training set, in normalized image coordinates. The final scale is 1.000, which is included by default. The box confidence threshold was 0.01 and the intersection over union threshold to match anchors to box labels was 0.3. The non-maximum suppression (NMS) threshold was 0.45 and the number of detected boxes to keep after NMS was 50.

The backbone and head were assigned initial learning rates of 0.002 and 0.02, respectively. Learning rates were annealed according to a cosine decay schedule. The model was trained for 30 epochs to minimize the loss function from the original SSD paper, which has a regression component for bounding box offsets and a classification component for distinguishing objects and background [103]. The weights corresponding to the epoch with the lowest validation loss were retained for test set evaluation.

Block index	Feature map dimensions ($w \times h \times c$)
1	$32 \times 32 \times 16$
3	$16 \times 16 \times 24$
6	$8 \times 8 \times 40$
9	$4 \times 4 \times 96$
12	$4 \times 4 \times 576$

Table G.1: MobileNetV3Small block indices and the corresponding dimensions of the feature maps that they output, given an input of size $128 \times 128 \times 3$.

Appendix H

Data Augmentation Leave-one-out Analysis Statistical Testing

As outlined in Section 6.4.1, statistical testing was performed to detect differences between pretrained models trained using an ablated version of the StandardAug and AugUS-O pipelines and baseline models pretrained on the original pipelines. Each ablated version of the pipeline was missing one transformation from the data augmentation pipeline. Ten-fold cross-validation conducted on the training set provided 10 samples of test AUC metrics for both the A-line versus B-line (`AB`) and pleural effusion (`PE`) binary classification tasks. The samples were taken as a proxy for test-time performance for linear classifiers trained on each of the above downstream tasks.

To determine whether the mean test AUC for each ablated model was different from the baseline model, hypothesis testing was conducted. The model pretrained using the original pipeline was the control group, while the models pretrained using ablated versions of the pipeline were the experimental groups. First, Friedman's test [48] was conducted to determine if there was any difference in the mean test AUC among the baseline and ablated models. We selected a nonparametric multiple comparisons test because of the lack of assumptions regarding normality or homogeneity of variances. Each collection had 10 samples. Table H.1 details the results of Friedman's test for each pipeline and classification task. Friedman's test detected differences among the collection of test AUC for both classification tasks with the StandardAug pipeline. Only the `AB` task exhibited significant differences for the AugUS-O pipeline.

When the null hypothesis of the Friedman test was rejected, post-hoc tests were conducted to determine whether any of the test AUC means in the experimental groups were

	AB		PE	
Pipeline	F_r Statistic	p-value	F_r Statistic	p-value
StandardAug	30.73	0.000*	36.51	0.000*
AugUS-O	76.43	0.000*	18.91	0.091

* Statistically significant at $\alpha = 0.05$.

Table H.1: Friedman test statistics (F_r) and p-values for mean cross-validation test AUC attained by models pretrained using an entire data augmentation pipeline and ablated versions of it.

significantly different than the control group. The Wilcoxon Sign-Rank Test [165] was designated as the post-hoc test, due to its absence of any normality assumptions. Note that for each pipeline, n comparisons were performed, where n is the number of transformations within the pipeline. The Holm-Bonferroni correction [70] was applied to keep the family-wise error rate at $\alpha = 0.05$ for each pipeline/task combination. Results of the post-hoc tests are given in Table H.2. No post-hoc tests were performed for the AugUS-O pipeline evaluated on the PE task because the Friedman test revealed no significant differences.

Pipeline	Comparison	AB		PE	
		T	p-value	T	p-value
StandardAug	*B00*	0	0.002*	0	0.002*
	B01	6	0.027	21	0.557
	B02	1	0.004*	3	0.010*
	B03	19	0.432	10	0.084
	B04	9	0.064	5	0.020
	B05	15	0.232	10	0.084
AugUS-O	*U00*	18	0.375	-	-
	U01	8	0.049	-	-
	U02	0	0.002*	-	-
	U03	0	0.002*	-	-
	U04	12	0.131	-	-
	U05	9	0.064	-	-
	U06	13	0.160	-	-
	U07	13	0.160	-	-
	U08	1	0.004*	-	-
	U09	1	0.004*	-	-
	U10	23	0.695	-	-
	U11	0	0.002*	-	-

* Statistically significant at family-wise error rate of 0.05.

Table H.2: Test statistics (T) and p-values obtained from the Wilcoxon Sign-Rank post-hoc tests that compared linear classifiers trained with ablated models' features to a control linear classifier trained on the baseline model. Experimental groups are identified according to the left-out transformation, as defined in Tables II and III.

Appendix I

Additional Positive Pair Examples

Figures I.1, I.2, and I.3 provide several examples of positive pairs produced by the StandardAug, AugUS-O, and AugUS-D pipelines, respectively. Each figure shows original images from LUSData, along with two views of each image that were produced by applying stochastic data augmentation twice to the original images.

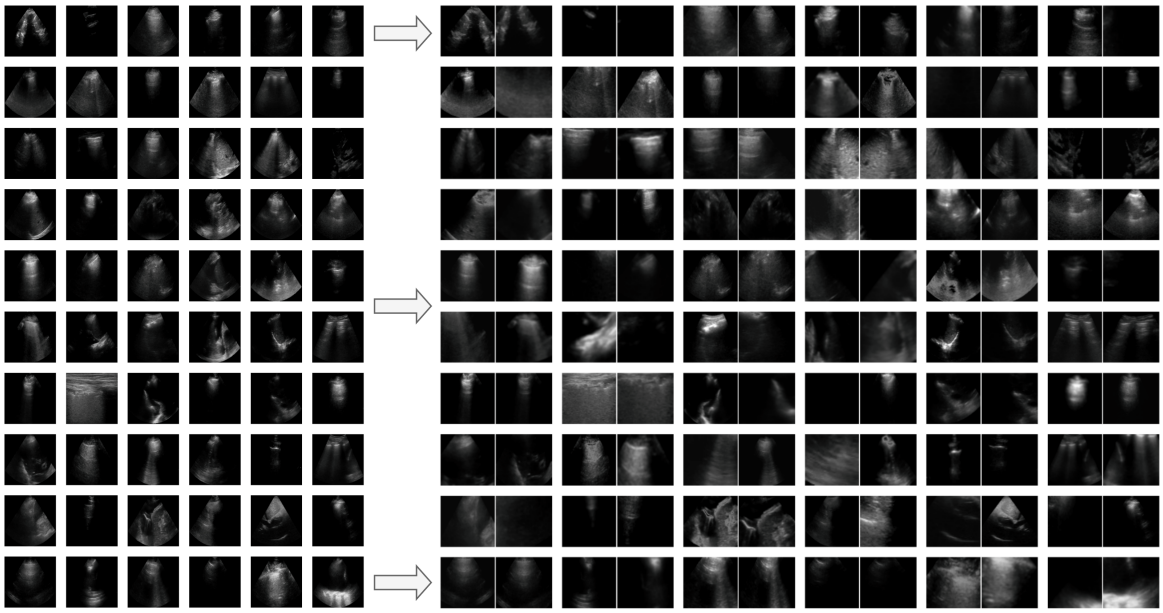


Figure I.1: Examples of lung ultrasound images (left) and corresponding positive pairs produced using the StandardAug pipeline (right).

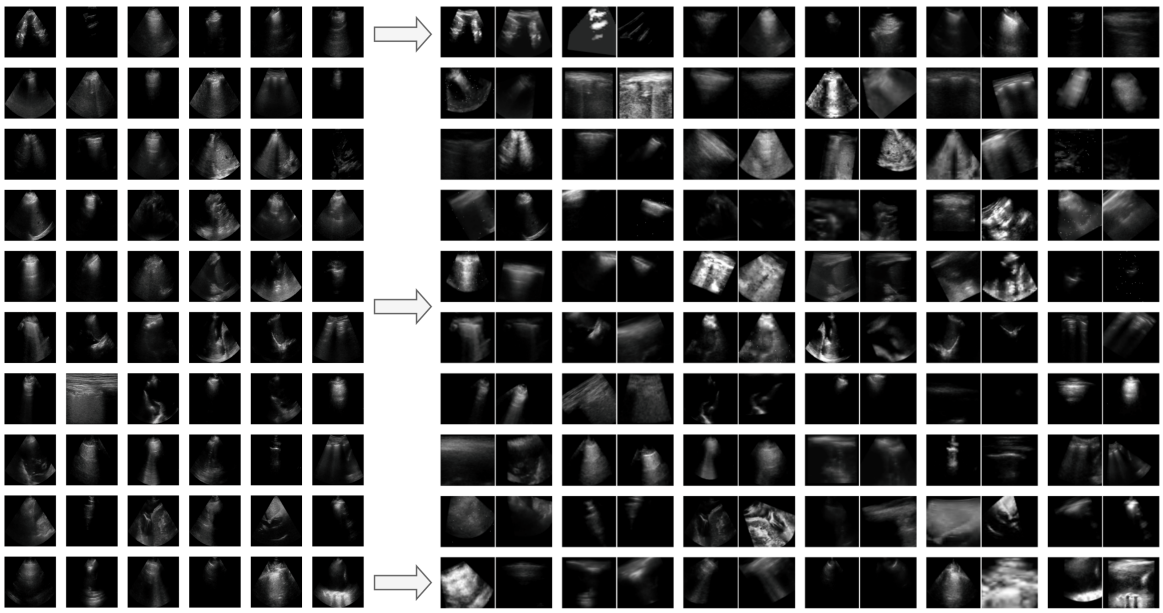


Figure I.2: Examples of lung ultrasound images (left) and corresponding positive pairs produced using the AugUS-O pipeline (right).

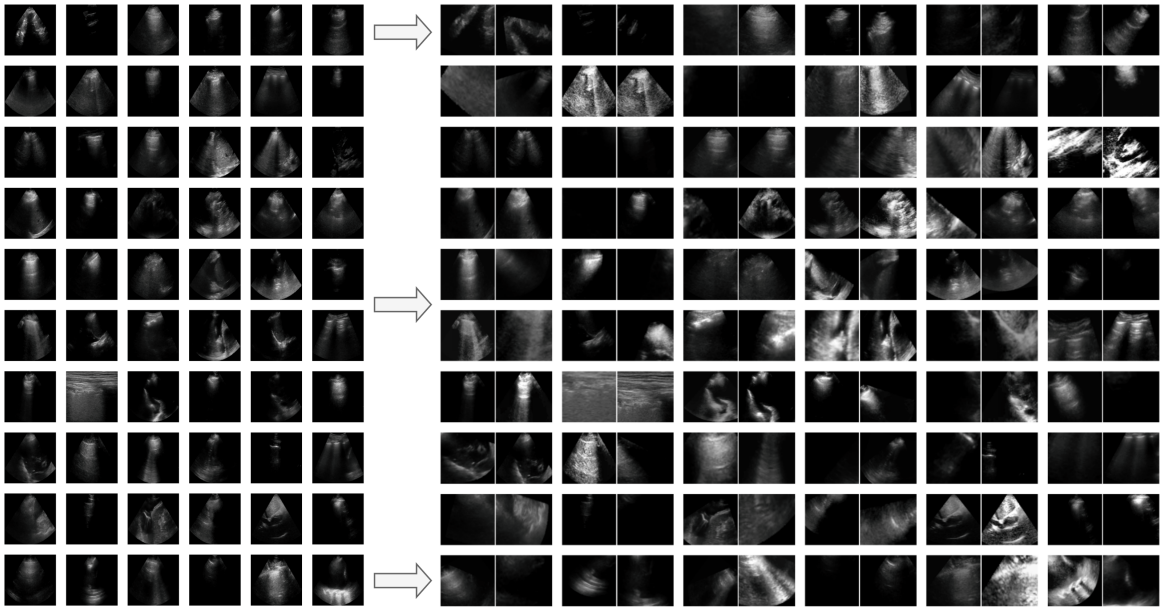


Figure I.3: Examples of lung ultrasound images (left) and corresponding positive pairs produced using the AugUS-D pipeline (right).

Appendix J

Data Augmentation Comparisons with ResNet18 Backbone

As outlined in Section 5.4.1, MobileNetV3Small [71] was selected as the feature extractor for all experiments in this study, primarily due to its suitability for lightweight inference in edge deployment scenarios. It has also been used in prior work for similar tasks [155, 157]. However, we wanted to determine if similar trends in the results held for an alternate backbone architecture with greater capacity. Larger CNN backbones are frequently used for self-supervised pretraining.

We repeated the experiments in Sections 6.4.2 and 6.4.3 using ResNet18 [68] as the feature extractor, which is a more common architecture. Note that ResNet18's capacity is far greater than that of MobileNetV3Small — the former has approximately 11 200 000 trainable parameters, while the latter has about 927 000. We applied the same architecture for the projector head that was used for MobileNetV3Small experiments, which was a 2-layer multilayer perceptron with 576 nodes comprising each layer.

ResNet18 feature extractors were pretrained using virtual machines equipped with an Intel Silver 4216 Cascade Lake CPU at 2.1 GHz and a Nvidia Tesla V100 GPU with 12 GB of VRAM. Supervised learning was conducted using a virtual machine equipped with an Intel E5-2683 v4 Broadwell CPU at 2.1 GHz and a Nvidia Tesla P100 GPU.

Table J.1 displays the performance of pretrained ResNet18 backbones when evaluated on the LUSData test set for the `AB`, `PE`, and `COVID` tasks. Most fine-tuned models exhibited overfitting, likely due to the architecture's capacity. Notably, the pretrained models tended to exhibit overfitting when fine-tuned, while the fully supervised ImageNet-pretrained model achieved strong performance on `AB` and `PE`. Linear classifiers trained on

frozen pretrained backbones' output features generally achieved the strongest performance and avoided overfitting. Consistent with MobileNetV3Small, the use of cropping-based augmentation pipelines translated to greater performance on `AB` and `PE` in the linear evaluation setting. In the fine-tuning setting, the backbone pretrained using AugUS-O exhibited markedly poorer performance on `AB` but achieved the greatest test AUC on `PE`. On the `COVID` test set, the backbone pretrained using the AugUS-O pipeline led to the greatest average AUC in both the linear and fine-tuning settings.

The fine-tuned backbones and linear classifiers were also evaluated on the external test set for the `AB` and `PE` tasks. Again, linear classifiers with pretrained backbones mostly achieved greater performance than fine-tuned classifiers. Backbones pretrained with the StandardAug pipeline achieved the greatest external test set AUC for the `AB` task. However, those pretrained with AugUS-O performed comparably with StandardAug for the `PE` task under the linear evaluation setting but achieved the greatest AUC in the fine-tuning setting.

Two key observations were drawn from these experiments. First, the low-capacity MobileNetV3Small backbones achieved similar performance to the high-capacity ResNet18 backbones for these LUS tasks when their weights were frozen (i.e., linear classifiers). Second, with a higher-capacity backbone, linear classifiers trained on the features outputted by SSL-pretrained backbones often achieved greater performance than fine-tuning using despite the weight initialization strategy. The opposite trend was observed for backbones initialized with ImageNet-pretrained weights.

Train Setting	Task	Weights	Pipeline	Accuracy	Precision	Recall	AUC
LC	AB	SimCLR	StandardAug	**0.932**	0.929	**0.845**	**0.968**
		SimCLR	AugUS-O	0.916	0.923	0.792	0.954
		SimCLR	AugUS-D	0.931	**0.951**	0.819	0.966
		ImageNet	-	0.878	0.838	0.749	0.933
	PE	SimCLR	StandardAug	0.781	0.772	0.777	**0.879**
		SimCLR	AugUS-O	**0.794**	**0.806**	0.759	0.878
		SimCLR	AugUS-D	0.779	0.739	**0.840**	0.873
		ImageNet	-	0.732	0.693	0.799	0.823
	COVID	SimCLR	StandardAug	0.537	0.538	0.496	**0.827**
		SimCLR	AugUS-O	**0.567**	**0.438**	**0.515**	0.782
		SimCLR	AugUS-D	0.487	0.353	0.447	0.789
		ImageNet	-	0.527	0.334	0.476	0.699
FT	AB	SimCLR	StandardAug	0.882	0.796	0.824	0.929
		SimCLR	AugUS-O	0.375	0.328	**0.990**	0.867
		SimCLR	AugUS-D	0.498	0.377	0.979	0.907
		Random	-	**0.919**	**0.898**	0.830	0.960
		ImageNet	-	0.892	0.778	0.906	**0.961**
	PE	SimCLR	StandardAug	0.692	0.643	0.817	0.795
		SimCLR	AugUS-O	0.773	**0.928**	0.576	**0.887**
		SimCLR	AugUS-D	0.654	0.601	**0.852**	0.755
		Random	-	0.720	0.684	0.784	0.815
		ImageNet	-	**0.778**	0.797	0.727	0.852
	COVID	SimCLR	StandardAug	0.418	**0.451**	0.447	0.719
		SimCLR	AugUS-O	**0.564**	0.434	**0.511**	**0.723**
		SimCLR	AugUS-D	0.431	0.346	0.376	0.631
		Random	-	0.372	0.277	0.354	0.629
		ImageNet	-	0.549	0.433	0.496	0.719

Table J.1: Test set performance for linear classification (LC) and fine-tuning (FT) experiments with the COVID task using pretrained ResNet18 backbones. For COVID, metrics are averages across all four classes. The best observed metrics in each experimental setting are in **boldface**.

Train Setting	Task	Initial Weights	Pipeline	Accuracy	Precision	Recall	AUC
LC	AB	SimCLR	StandardAug	**0.762**	**0.956**	**0.581**	**0.877**
		SimCLR	AugUS-O	0.713	0.939	0.494	0.826
		SimCLR	AugUS-D	0.739	0.945	0.542	0.862
		ImageNet	-	0.659	0.904	0.402	0.785
	PE	SimCLR	StandardAug	**0.797**	0.884	0.784	**0.887**
		SimCLR	AugUS-O	0.786	**0.911**	0.737	**0.887**
		SimCLR	AugUS-D	0.772	0.818	**0.826**	0.851
		ImageNet	-	0.701	0.790	0.723	0.776
FT	AB	SimCLR	StandardAug	0.767	**0.951**	0.594	**0.880**
		SimCLR	AugUS-O	0.597	0.576	**0.924**	0.728
		SimCLR	AugUS-D	0.697	0.666	0.867	0.816
		Random	-	0.744	0.919	0.576	0.851
		ImageNet	-	**0.771**	0.855	0.688	0.857
	PE	SimCLR	StandardAug	0.612	0.791	0.532	0.700
		SimCLR	AugUS-O	0.678	**0.972**	0.509	**0.897**
		SimCLR	AugUS-D	0.703	0.752	**0.797**	0.755
		Random	-	0.743	0.845	0.731	0.809
		ImageNet	-	**0.764**	0.897	0.712	0.854

Table J.2: External test set performance for linear classifiers (LC) and fine-tuned models (FT). The best observed metrics in each experimental setting are in **boldface**.

Appendix K

Data Augmentation Label Efficiency Statistical Testing

For each of the AB and PE tasks, there were five experimental conditions: SimCLR pretraining with the StandardAug pipeline, SimCLR pretraining with the AugUS-O pipeline, SimCLR pretraining with the AugUS-D pipeline, ImageNet weight initializations, and random weight initialization. The population consisted of 20 subsets of the training set that were split randomly by patient. The same splits were used across all conditions, reflecting a repeated measures design. The experiment was repeated separately for the AB and the PE task.

The Friedman Test Statistic (F_r) was 75.44 for AB, with a p-value of 0. For PE, the $F_r = 45.44$, and the p-value was 0. As such, the null hypothesis was rejected for both cases, indicating the existence of differences among the mean test AUC across conditions. The Wilcoxon Sign-Rank test was performed as a post-hoc test between each pair of populations. Tables K.1 and K.2 provide all Wilcoxon Test Statistics, along with p-values and differences of the medians between conditions. The Bonferroni correction was applied to the p-values to keep the family-wise error rate to $\alpha = 0.05$. Statistically significant comparisons are indicated.

Comparison	T	p-value	Δ
Random / ImageNet	3	$9.4 \times 10^{-5^*}$	-0.058
Random / StandardAug	0	$1.9 \times 10^{-5^*}$	0.044
Random / AugUS-O	12	$1.3 \times 10^{-3^*}$	0.019
Random / AugUS-D	0	$1.9 \times 10^{-5^*}$	0.042
ImageNet / StandardAug	0	$1.9 \times 10^{-5^*}$	0.100
ImageNet / AugUS-O	0	$1.9 \times 10^{-5^*}$	0.078
ImageNet / AugUS-D	0	$1.9 \times 10^{-5^*}$	0.100
StandardAug / AugUS-O	0	$1.9 \times 10^{-5^*}$	-0.024
StandardAug / AugUS-D	61	1.0	-0.002
AugUS-O / AugUS-D	0	$1.9 \times 10^{-5^*}$	0.022

* Statistically significant at family-wise error rate of 0.05.

Table K.1: Test statistics (T) and p-values obtained from the Wilcoxon Sign-Rank post-hoc tests comparing LUSData test AUC on `AB` for classifiers trained on subsets of the training set. For comparison a/b, $\Delta := \text{median}(b) - \text{median}(a)$. The displayed $p-$values have been adjusted according to the Bonferroni correction to control the family-wise error rate.

Comparison	T	p-value	Δ
Random / ImageNet	1	$3.8 \times 10^{-5^*}$	-0.127
Random / StandardAug	31	$4.2 \times 10^{-2^*}$	0.025
Random / AugUS-O	16	$3.2 \times 10^{-3^*}$	0.030
Random / AugUS-D	5	$1.9 \times 10^{-4^*}$	0.036
ImageNet / StandardAug	1	$3.8 \times 10^{-5^*}$	0.152
ImageNet / AugUS-O	0	$1.9 \times 10^{-5^*}$	0.157
ImageNet / AugUS-D	1	$3.8 \times 10^{-5^*}$	0.163
StandardAug / AugUS-O	57	7.6×10^{-1}	0.005
StandardAug / AugUS-D	53	5.3×10^{-1}	0.011
AugUS-O / AugUS-D	78	1	0.006

* Statistically significant at family-wise error rate of 0.05.

Table K.2: Test statistics (T) and p-values obtained from the Wilcoxon Sign-Rank post-hoc tests comparing LUSData test AUC on `PE` for classifiers trained on subsets of the training set. For comparison a/b, $\Delta := \text{median}(b) - \text{median}(a)$. The displayed $p-$values have been adjusted according to the Bonferroni correction to control the family-wise error rate.

Glossary

A-line A horizontal artefact seen on lung ultrasound imaging that is created by the reverberation of sound waves between the pleural line and the probe. It is suggestive of normally aerated lung tissue. Visual examples are provided in Figure 2.4b. 12, 43, 81, 99, 123

B-line A vertical "comet-tail" artefact seen on lung ultrasound imaging that is created by sound reverberating in regions of fluid-filled lung tissue. While they may occasionally be seen in healthy individuals, they are typically indicative of diseased lung tissue. Visual examples are provided in Figure 2.4b. 12, 43, 81, 100, 123

B-mode Also known as "brightness mode", a two-dimensional ultrasound image. It is typically acquired as a stream of two-dimensional grayscale images. Pixel locations correspond to distance from the probe, which is at the top of the image, by convention. The brightness of each pixel corresponds to the amplitude of sound waves reflected off of tissues that are received by the probe's transducer. Figure 2.2 depicts an example of a B-mode ultrasound image. 7, 41, 56, 96, 123

backbone A term often used for "feature extractor". 20, 126

barcode sign A pattern visible on M-mode lung ultrasound that indicates the absence of the lung sliding sign. It appears as stacked, horizontal lines below the pleural line. Visual examples are provided in Figure 2.4c. 8, 57, 127

capacity In the context of machine learning, the capacity of a neural network refers to the variety of functions that it can approximate. Capacity is influenced by architectural attributes of the neural network, such as the number of layers, the number of learnable parameters, and the choice of activation functions. 62, 113, 128, 187

contrastive learning A category of joint embedding self-supervised learning where backbones are trained to output similar representations for positive pairs and dissimilar representations for negative pairs. 23, 49, 57, 123

decoder A neural network that transforms latent representations to raw input space. 21

downstream task A machine learning task of interest (typically supervised learning) that is intended to be learned using a neural network that is initialized using pretrained weights. 20, 57, 78, 99, 124

echocardiography Ultrasound examination of the heart to assess cardiac structure and function. 9, 78, 132

embedding A vector output of a deep neural network in the context of self-supervised pretraining. Embeddings typically comprise the inputs to loss functions in joint embedding self-supervised learning. Ordinarily, the embeddings are the final output of a deep neural network that consists of a backbone and an additional neural network (often referred to as a projector) that are connected in series. 23

encoder A term often used for "feature extractor". 20

feature extractor A neural network that ingests raw inputs and outputs feature vectors that contain salient information from the inputs. 20, 41, 123

fully supervised model A machine learning algorithm trained to solve a supervised learning task that did not undergo self-supervised pretraining. 30, 46, 65, 83

information collapse A degenerate solution to non-contrastive learning in which backbones learn to indiscriminately output the zero vector as a representation for all inputs. 25

joint embedding self-supervised learning A family of self-supervised learning methods where the pretext task is to produce similar vector representations for semantically related pairs of images. In a typical training step, two batches of corresponding examples are mapped to a common embedding space. The i^{th} example of both batches are intended to be semantically related. Figure 2.9 provides a visual overview of these methods. 23, 42, 77, 96, 123

lung sliding A normal finding on lung ultrasound that consists of the apposition of the visceral and parietal pleurae. It can be visualized on B-mode videos or M-mode images of the lung. The absence of lung sliding is suggestive of pneumothorax. Visual examples of lung sliding are provided in Figure 2.4c. 8, 56, 82, 123

M-mode Also known as "motion mode", a form of ultrasound imaging that depicts the evolution of a single-dimensional scan line over time. The brightness of each pixel corresponds to the amplitude of sound waves received by the probe's transducer at a point in time that were reflected from tissues at a specific distance. Figure 2.2 depicts an example of a M-mode mode ultrasound image. 8, 56, 123

negative pair In joint embedding self-supervised learning, any two inputs that do not satisfy the pairwise relationship. 23, 75, 124

non-contrastive learning A category of joint embedding self-supervised learning where backbones are trained to output similar representations for positive pairs. Typically, they integrate strategies for avoiding information collapse. 23, 42, 57, 123

pairwise relationship In joint embedding self-supervised learning, the pairwise relationship is satisfied for any two examples that are similar to each other. The relationship is designed to reflect the characteristics of the data distribution. 23, 75, 98

parenchymal view An ultrasound view of the lung that reveals information about the lung tissue. It is acquired by placing the probe on the anterior or anterolateral chest, between adjacent ribs. Visual examples are provided in Figure 2.4a. 10, 43, 56, 99

parietal pleura The outer layer of the lining of the lungs. It is a membrane that adheres to the interior of the chest wall. During normal respiration, it slides along the visceral pleura. 10, 56

pleura A layer of the lining of the lungs. There are two such layers: the parietal (outer) pleura and the visceral (inner) pleura. 10, 14, 56

pleural effusion A collection of fluid between the visceral and parietal pleurae that can be seen in lung ultrasound examinations. While small pleural effusions may occur in healthy individuals, they are typically pathological. Visual examples are provided in Figure 2.4d. 15, 43, 100, 123

pleural line An object visible on parenchymal lung ultrasound images that corresponds to the pleurae of the lung. It usually appears near the top of the image. Discontinuities in the pleural line are often caused by ribs, which are superficial to the pleurae. 8, 10, 100

pleural view An ultrasound view of the lung that reveals information about the pleural space external to the lung, between the visceral and parietal pleurae. It is acquired by placing the probe on the posterolateral chest. Visual examples are provided in Figure 2.4a. 10, 43, 99

positive pair In joint embedding self-supervised learning, any two inputs that satisfy the pairwise relationship. 23, 75, 96, 124

pretext task A supervised learning task that a feature extractor learns to perform during self-supervised pretraining. Typically, the targets for the pretext task are readily computable from the examples in the dataset. Annotations made by humans are not employed as targets for pretext tasks. 18, 60, 123

projector In the context of joint embedding self-supervised learning, the projector is a neural network that maps representations to embeddings. Backbone and projector networks are typically connected in series and trained end-to-end during self-supervised pretraining. Ordinarily, the architecture of the projector is a multi-layer perceptron. 23, 40, 62, 78, 107

representation A vector output of a deep neural network backbone. Useful representations contain salient information about the backbone's input. 20, 23

representation learning A form of machine learning in which algorithms are trained to distill salient information from examples in a compact form. 19, 42

seashore sign A grainy pattern visible on M-mode lung ultrasound below the pleural line that indicates the presence of the lung sliding sign. Visual examples are provided in Figure 2.4c. 8, 57, 127

self-supervised pretraining The process of training a neural network using a self-supervised learning method, with the intention of using the resulting weights in the initialization of a neural network for a separate learning task. 18, 21, 123

stratosphere sign Another term for "barcode sign". 57

visceral pleura The inner layer of the lining of the lungs. It is a membrane that wraps around the lungs. During normal respiration, it slides along the parietal pleura. 10, 56